

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Pelaksanaan kerja magang dilakukan di PT Matica Inovasi Edukasi sebagai *Mobile Application Developer Intern*, dengan fokus utama pada pengembangan aplikasi *mobile* berbasis Android. Tanggung jawab yang dijalankan mencakup pengembangan aplikasi *Learning Management System* (LMS) menggunakan bahasa pemrograman Kotlin melalui Android Studio sebagai lingkungan pengembangan utama. Sistem *back-end* aplikasi ini terintegrasi dengan Firebase dan Supabase, sesuai dengan fungsi masing-masing layanan.

Proses pembuatan aplikasi dilakukan di bawah bimbingan langsung dari *supervisor* teknis dan Senior IT, yang memberikan arahan serta supervisi selama kegiatan berlangsung. Koordinasi kerja dilaksanakan secara daring melalui platform komunikasi seperti *WhatsApp* dan *Google Meet* saat *Work From Home* (WFH), serta melalui diskusi langsung di kantor selama *Work From Office* (WFO).

Beberapa fitur utama yang berhasil dikembangkan meliputi fitur riwayat kehadiran (*Attendance History*) untuk orang tua dan siswa, sistem asesmen (*Assessment*) untuk guru dan orang tua, serta fitur unggah materi pembelajaran yang dapat diakses oleh guru, murid, dan orang tua. Pengelolaan kode dilakukan menggunakan GitHub, dengan penerapan sistem pengembangan berbasis *branch* dan *pull request*. Penamaan *branch* disesuaikan dengan nama masing-masing pengembang, dan proses *code review* dilakukan secara berkala untuk memastikan kualitas serta konsistensi kode sebelum integrasi ke repositori utama.

3.2 Tugas yang Dilakukan

Proses magang yang dilaksanakan di PT Matica Inovasi Edukasi berfokus pada pengembangan sistem *Learning Management System* (LMS) berbasis Android. Ruang lingkup pekerjaan mencakup perancangan antarmuka pengguna, implementasi logika fungsionalitas, serta integrasi dengan layanan *back-end*. Dalam proyek ini, peran yang dijalankan adalah sebagai *Mobile Application Developer*, yang bertanggung jawab dalam pengembangan aplikasi secara menyeluruh, baik dari sisi *front-end* maupun *back-end*.

Aplikasi dikembangkan menggunakan Android Studio sebagai *Integrated*

Development Environment (IDE), dengan Kotlin sebagai bahasa pemrograman utama. Untuk mendukung fungsi backend, Firebase dimanfaatkan sebagai layanan autentikasi dan basis data utama yang menangani penyimpanan informasi pengguna, data kehadiran, serta asesmen. Selain itu, Supabase turut diintegrasikan sebagai layanan penyimpanan eksternal untuk mendukung fitur unggah dokumen. Layanan ini dikonfigurasi agar berjalan aman melalui integrasi dengan *Firebase Authentication*, sehingga akses file dapat dibatasi sesuai peran pengguna.

Lingkup pekerjaan juga mencakup perancangan antarmuka serta logika *back-end* untuk seluruh halaman pengguna *Parent* dan *Student*, seperti halaman *login*, *dashboard*, presensi, asesmen, materi pembelajaran, latihan soal, hingga fitur pencarian video pembelajaran. Di samping itu, dirancang pula halaman utama untuk pemilihan peran pengguna (*mainpage*), beserta sistem navigasi dinamis berdasarkan *role*.

Seluruh proses kerja dilakukan secara bertahap, dimulai dari desain antarmuka pada Figma, inisialisasi proyek, pengembangan fungsionalitas inti, hingga tahap finalisasi dan pengujian menyeluruh. Setiap tahapan bertujuan memastikan kesesuaian fitur dengan kebutuhan pengguna serta kestabilan sistem yang dihasilkan.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan kerja magang diuraikan seperti pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu selama periode magang

Minggu Ke-	Pekerjaan yang dilakukan
1	Melakukan pembahasan awal mengenai sistem aplikasi LMS yang akan dikembangkan, termasuk identifikasi jumlah pengguna (<i>admin</i> , <i>teacher</i> , <i>student</i> , dan <i>parent</i>) serta fitur utama untuk masing-masing <i>role</i> . Hasil diskusi menetapkan tanggung jawab yang diemban mencakup pembuatan tampilan pemilihan <i>role</i> (<i>mainpage</i>) serta keseluruhan halaman untuk <i>student</i> dan <i>parent</i> .
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilaksanakan setiap minggu selama periode magang (lanjutan)

Minggu Ke-	Pekerjaan yang dilakukan
2	Melakukan riset referensi desain antarmuka menggunakan Figma dan mendiskusikan pilihan skema warna aplikasi dengan tim pengembang. Referensi desain dikumpulkan dari berbagai sumber untuk memastikan konsistensi tampilan visual yang sesuai, kemudian disampaikan kepada <i>supervisor</i> untuk mendapatkan persetujuan arah desain.
3	Mendesain halaman <i>Parent Page</i> yang terdiri dari <i>Parent Login</i> , <i>Home Parent</i> , <i>Assessment Parent</i> , <i>Absent Parent</i> , dan <i>Send File</i> . <i>Assessment</i> menampilkan daftar anak serta detail asesmen seperti mata pelajaran, topik, dan performa. Fitur <i>Absent</i> menampilkan histori kehadiran anak, dan <i>Send File</i> untuk mengunggah dokumen ke guru. Semua tombol diuji agar berfungsi dari <i>mainpage</i> .
4	Mendesain halaman <i>Student Page</i> yang meliputi <i>Student Login</i> , <i>Home Student</i> , <i>Absent Student</i> , <i>Feedback Page</i> , <i>Online Class</i> , <i>Notes</i> , <i>Exercise</i> , dan <i>Explore Video</i> . <i>Explore Video</i> memiliki fitur <i>search bar</i> dan pemilihan kategori berdasarkan mata pelajaran. Semua komponen diatur agar berjalan fungsional dengan tombol interaktif dari <i>mainpage</i> .
5	Melakukan pelaporan hasil desain Figma kepada <i>supervisor</i> dan merevisi sesuai umpan balik yang diberikan, termasuk penyesuaian warna, navigasi, dan komponen antarmuka agar konsisten serta sesuai harapan pengguna.
6	Menginisialisasi proyek menggunakan Android Studio dan menghubungkannya dengan Firebase untuk autentikasi serta penyimpanan data. Mengembangkan tampilan <i>RoleFragment</i> sebagai halaman awal pemilihan peran, dan memulai implementasi halaman <i>Parent Page</i> (<i>Login</i> , <i>Home</i> , <i>Assessment</i>) sesuai desain dengan menggunakan data <i>dummy</i> .
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilaksanakan setiap minggu selama periode magang (lanjutan)

Minggu Ke-	Pekerjaan yang dilakukan
7	Melanjutkan pembuatan <i>front-end Parent Page</i> , khususnya fitur <i>Send File</i> dan <i>Absent</i> . Fitur <i>Send File</i> dikembangkan agar dapat mengunggah dokumen ke guru, sementara fitur <i>Absent</i> menampilkan histori kehadiran anak secara dinamis. Semua halaman diuji fungsionalitas dan navigasinya.
8	Mengembangkan bagian awal halaman <i>Student Page</i> , seperti <i>Student Login</i> , <i>Home Student</i> , <i>Absent Student</i> , dan <i>Online Class</i> . Seluruh fitur dikembangkan menggunakan data <i>dummy</i> dan diuji untuk memastikan navigasi antar halaman berjalan sesuai alur.
9	Melanjutkan pembuatan halaman <i>Student Page</i> , termasuk <i>Feedback Page</i> (dengan sistem rating dan catatan), halaman <i>Notes</i> , <i>Exercise</i> , serta <i>Explore Video</i> . <i>Explore Video</i> mendukung pencarian berdasarkan kategori mata pelajaran. Seluruh halaman diuji menggunakan data <i>dummy</i> untuk memastikan kesesuaian alur dan tampilan.
10	Melakukan finalisasi dan revisi <i>front-end</i> berdasarkan hasil uji coba internal dan masukan dari <i>supervisor</i> . Penambahan fitur konfirmasi sebelum presensi serta peningkatan tampilan antarmuka dilakukan agar lebih intuitif dan konsisten.
11	Membahas <i>back-end framework</i> alternatif karena keterbatasan biaya <i>Firebase Storage</i> , dan diputuskan untuk menggunakan <i>Supabase</i> sebagai layanan penyimpanan file. Dilakukan pula implementasi sistem autentikasi berbasis email dan <i>password</i> pada <i>Firebase</i> .
12	Mengembangkan <i>back-end</i> untuk <i>Parent</i> dan <i>Student Page</i> , seperti menampilkan nama pengguna dengan imbuhan <i>Ms./Mr.</i> sesuai <i>gender</i> , menyimpan data kehadiran siswa, dan menampilkan histori kehadiran pada halaman <i>parent</i> . Menambahkan sistem presensi otomatis saat bergabung ke <i>Online Class</i> . Mengimplementasikan <i>toast message</i> sebagai notifikasi atas aksi yang dilakukan pengguna.
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilaksanakan setiap minggu selama periode magang (lanjutan)

Minggu Ke-	Pekerjaan yang dilakukan
13	Mengembangkan <i>back-end</i> untuk fitur <i>Feedback Student</i> , termasuk pencatatan <i>rating</i> , komentar, serta pencarian nama guru. <i>Parent Page</i> juga ditambahkan fitur untuk menampilkan data asesmen dari guru.
14	Mengembangkan <i>back-end</i> untuk fitur <i>Explore Video</i> , mencakup pencarian video, pemilihan kategori, dan tampilan personalisasi (<i>For You</i>). Semua fitur diuji agar berjalan dengan baik. Menambahkan notifikasi berbasis <i>Toast</i> menggunakan <i>makeText</i> untuk memberikan umpan balik kepada pengguna setelah suatu aksi dilakukan, baik dalam kondisi berhasil maupun gagal.
15	Mengintegrasikan Supabase sebagai penyimpanan file, serta menghubungkannya dengan Firebase untuk menyimpan dan membaca <i>link</i> file. Mengerjakan <i>back-end</i> untuk fitur <i>Send File Parent</i> , <i>Notes</i> serta <i>Exercise</i> pada <i>Student Page</i> .
16	Melakukan revisi tampilan <i>front-end</i> untuk meningkatkan daya tarik visual, menambahkan fitur notifikasi pada <i>Parent Page</i> , serta menambahkan fungsi <i>logout</i> . Uji coba sistem dilakukan untuk memastikan semua fitur berjalan stabil.

3.4 Analisis dan Perancangan Sistem

Pada tahap ini, dilakukan analisis dan perancangan sistem aplikasi *Learning Management System* (LMS) berbasis Android yang dikembangkan di PT Matica Inovasi Edukasi. Proses perancangan ini melibatkan penggunaan beberapa diagram yang digunakan untuk menggambarkan alur, interaksi, dan struktur sistem secara keseluruhan. Beberapa diagram yang digunakan untuk perancangan sistem ini antara lain *Usecase Diagram*, *Sequence Diagram*, *Activity Diagram*, dan *Class Diagram*. Setiap diagram memiliki fungsi dan kegunaan yang berbeda, namun secara keseluruhan, diagram-diagram ini membantu dalam merancang dan memetakan bagaimana sistem akan bekerja serta bagaimana interaksi pengguna dan komponen sistem terjadi.

Selain itu, untuk mendukung perancangan yang lebih detail, pada subbab ini juga akan dibahas Struktur Tabel yang menggambarkan bagaimana data akan disimpan dalam *database* aplikasi dan Hasil Implementasi yang menunjukkan hasil nyata dari implementasi sistem yang dirancang.

3.4.1 Use Case Diagram

Use Case Diagram adalah salah satu diagram dalam *Unified Modeling Language* (UML) yang menggambarkan interaksi antara aktor (pengguna atau sistem lain) dengan sistem yang sedang dikembangkan. Diagram ini digunakan untuk mendefinisikan fungsionalitas sistem dari perspektif pengguna, menunjukkan apa yang dapat dilakukan oleh pengguna dalam sistem tanpa menjelaskan bagaimana sistem tersebut diimplementasikan secara teknis. *Use Case Diagram* membantu dalam mengidentifikasi kebutuhan sistem dan memastikan bahwa semua fungsi yang diperlukan oleh pengguna telah dipertimbangkan dalam perancangan sistem [8][9].

Dalam konteks aplikasi LMS di PT Matica Inovasi Edukasi, *Use Case Diagram* digunakan untuk menggambarkan interaksi antara empat peran utama pengguna, yaitu *Admin*, *Teacher*, *Parent*, dan *Student* dengan sistem. Setiap aktor memiliki serangkaian *use case* yang mencerminkan fungsionalitas yang dapat diaksesnya, seperti *login*, presensi, pengiriman tugas, dan penilaian. Diagram ini memberikan gambaran umum tentang bagaimana setiap pengguna berinteraksi dengan sistem, membantu dalam perencanaan dan pembuatan fitur yang sesuai dengan kebutuhan masing-masing peran.

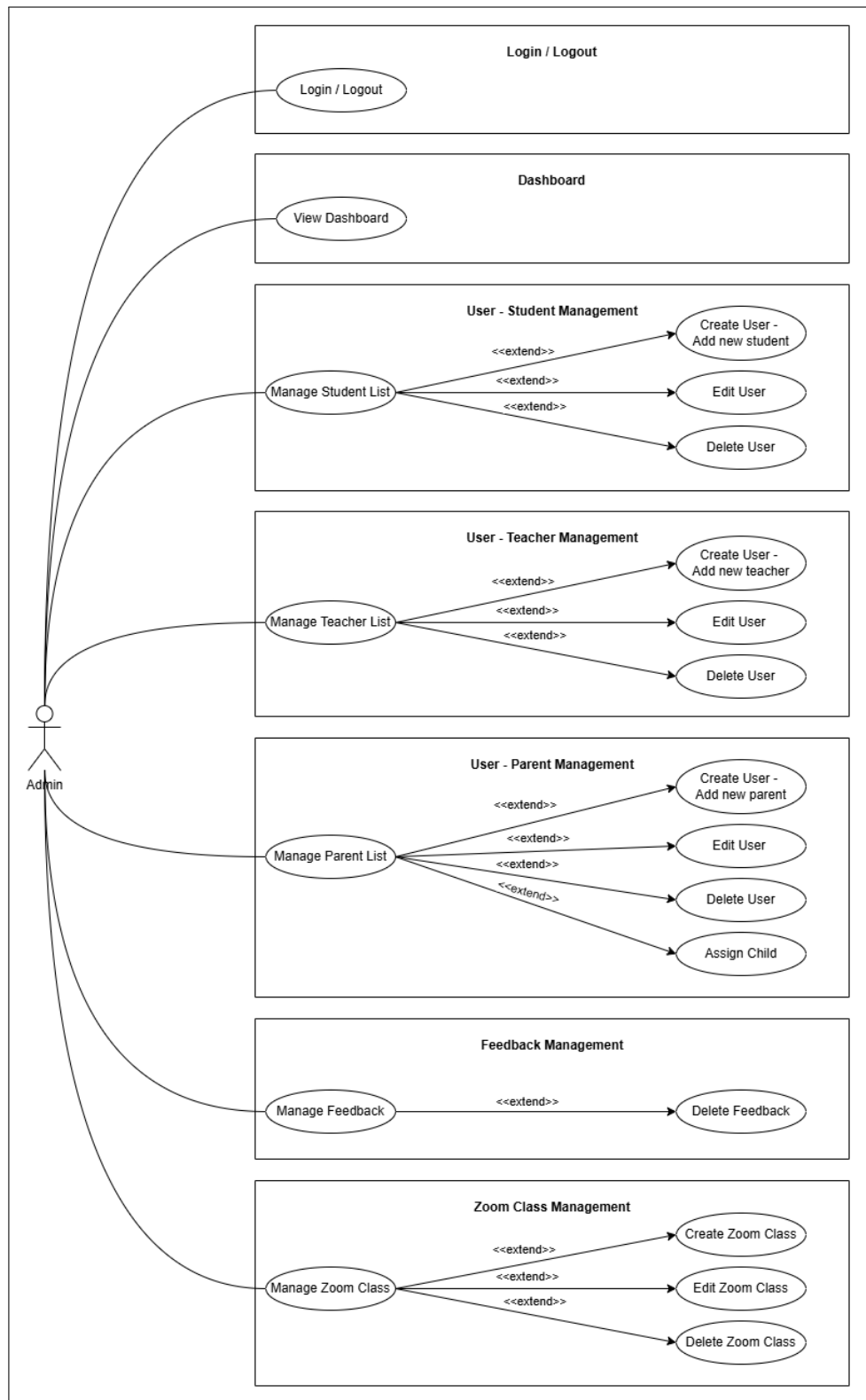
A Admin

Use Case Diagram Admin menggambarkan interaksi antara aktor *Admin* dengan sistem *Learning Management System (LMS)* , sebagaimana ditunjukkan pada Gambar 3.1. Aktor *Admin* memiliki akses penuh terhadap berbagai fitur manajemen pengguna dan konten dalam sistem. *Admin* dapat melakukan proses *login* dan *logout*, serta mengakses *dashboard* sebagai pusat navigasi utama sistem.

Diagram ini menunjukkan bahwa *Admin* memiliki peran penting dalam manajemen data pengguna, mencakup siswa, guru, dan orang tua. Setiap proses manajemen pengguna seperti menambah, mengubah, dan menghapus data diatur melalui fungsi-fungsi seperti *Manage Student List*, *Manage Teacher List*, dan *Manage Parent List*. Masing-masing fungsi tersebut diperluas (*extend*) dengan *use case* tambahan seperti *Create User*, *Edit User*, dan *Delete User*. Khusus untuk orang tua, terdapat tambahan fungsi *Assign Child* untuk mengaitkan siswa dengan akun orang tua.

Selain manajemen pengguna, *Admin* juga dapat mengelola umpan balik yang dikirimkan melalui fitur *Manage Feedback* yang memiliki ekstensi berupa tindakan *Delete Feedback*. *Admin* turut memiliki akses untuk mengelola kelas *online* melalui fitur *Manage Zoom Class*, termasuk membuat, mengedit, dan menghapus jadwal kelas virtual. Keseluruhan diagram ini mencerminkan peran sentral *Admin* dalam memastikan sistem LMS berjalan dengan baik, dengan struktur relasi *extend* yang mengindikasikan perluasan fungsi dari *use case* utama.





Gambar 3.1. Use Case Diagram Admin PT Matica Inovasi Edukasi

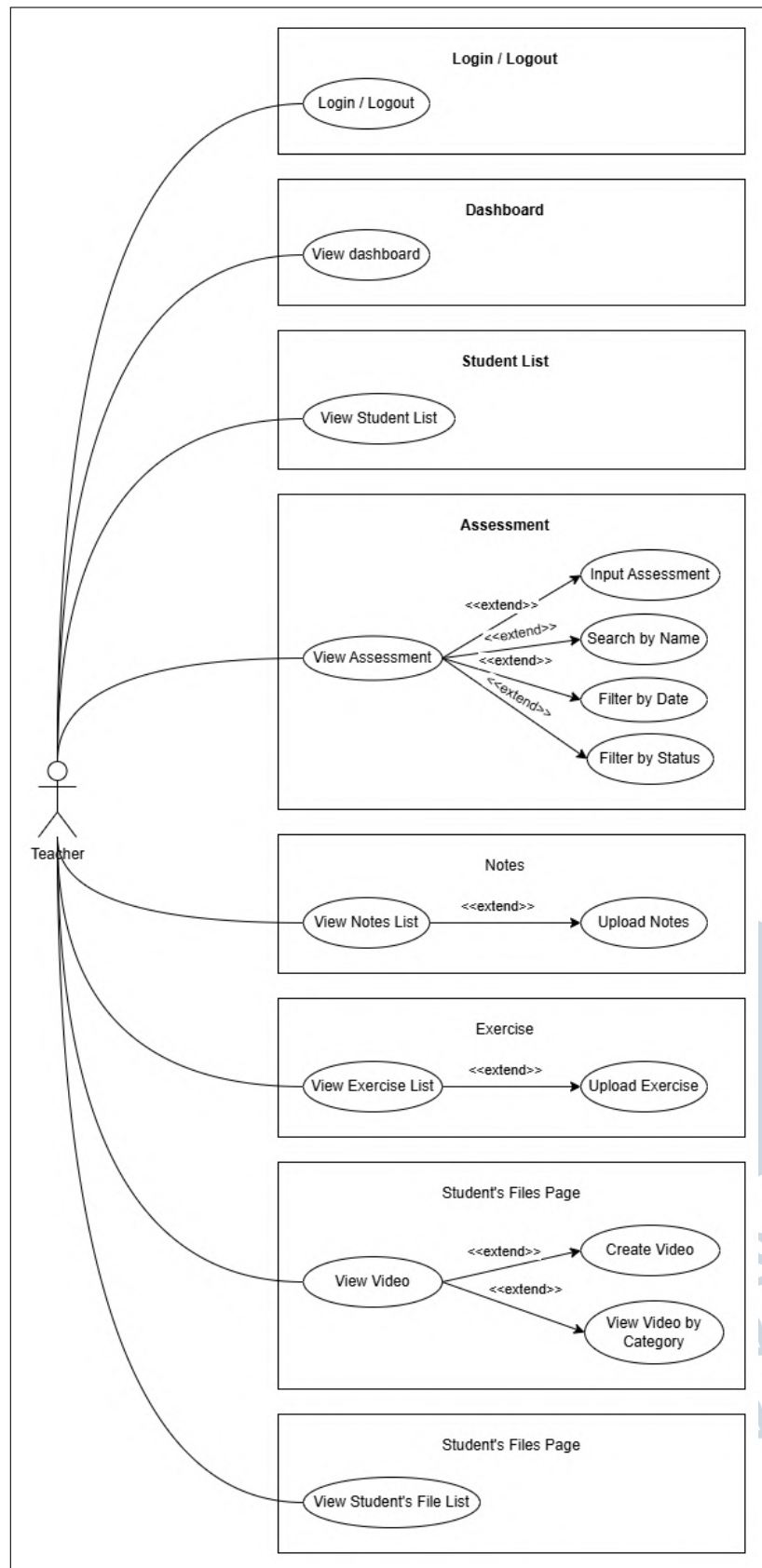
B Teacher

Use Case Diagram Teacher menggambarkan interaksi antara aktor *Teacher* (guru) dengan sistem *Learning Management System* (LMS) , sebagaimana ditunjukkan pada Gambar 3.2. Guru memiliki sejumlah hak akses terhadap fitur-fitur sistem yang berkaitan dengan pengelolaan siswa, materi pembelajaran, penilaian, dan konten multimedia. Diawali dengan proses *Login/Logout*, guru dapat mengakses *Dashboard* dan melihat daftar siswa melalui fitur *View Student List*.

Salah satu fitur utama yang dimiliki oleh guru adalah *View Assessment*, yang dilengkapi dengan beberapa fungsi tambahan seperti *Input Assessment*, *Search by Name*, *Filter by Date*, dan *Filter by Status* untuk mempermudah pencarian dan pengelompokan data penilaian siswa. Guru juga memiliki akses terhadap halaman catatan (*Notes*) dan latihan (*Exercise*) dengan kemampuan untuk melihat daftar yang tersedia serta mengunggah materi baru melalui fitur *Upload Notes* dan *Upload Exercise*.

Selain itu, guru dapat mengakses halaman file milik siswa dalam bentuk video atau dokumen lain. Melalui fitur *View Video*, guru dapat mengunggah konten pembelajaran dalam bentuk video (*Create Video*) serta menyaring video berdasarkan kategori (*View Video by Category*). Guru juga dapat melihat daftar file siswa serta mengakses file tersebut secara langsung menggunakan fitur *View Student's File List*. Secara keseluruhan, *use case diagram* ini menunjukkan peran guru dalam menyampaikan materi, melakukan asesmen, serta mengelola konten digital secara mandiri dan terintegrasi dalam sistem LMS.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.2. Use Case Diagram Teacher PT Matica Inovasi Edukasi

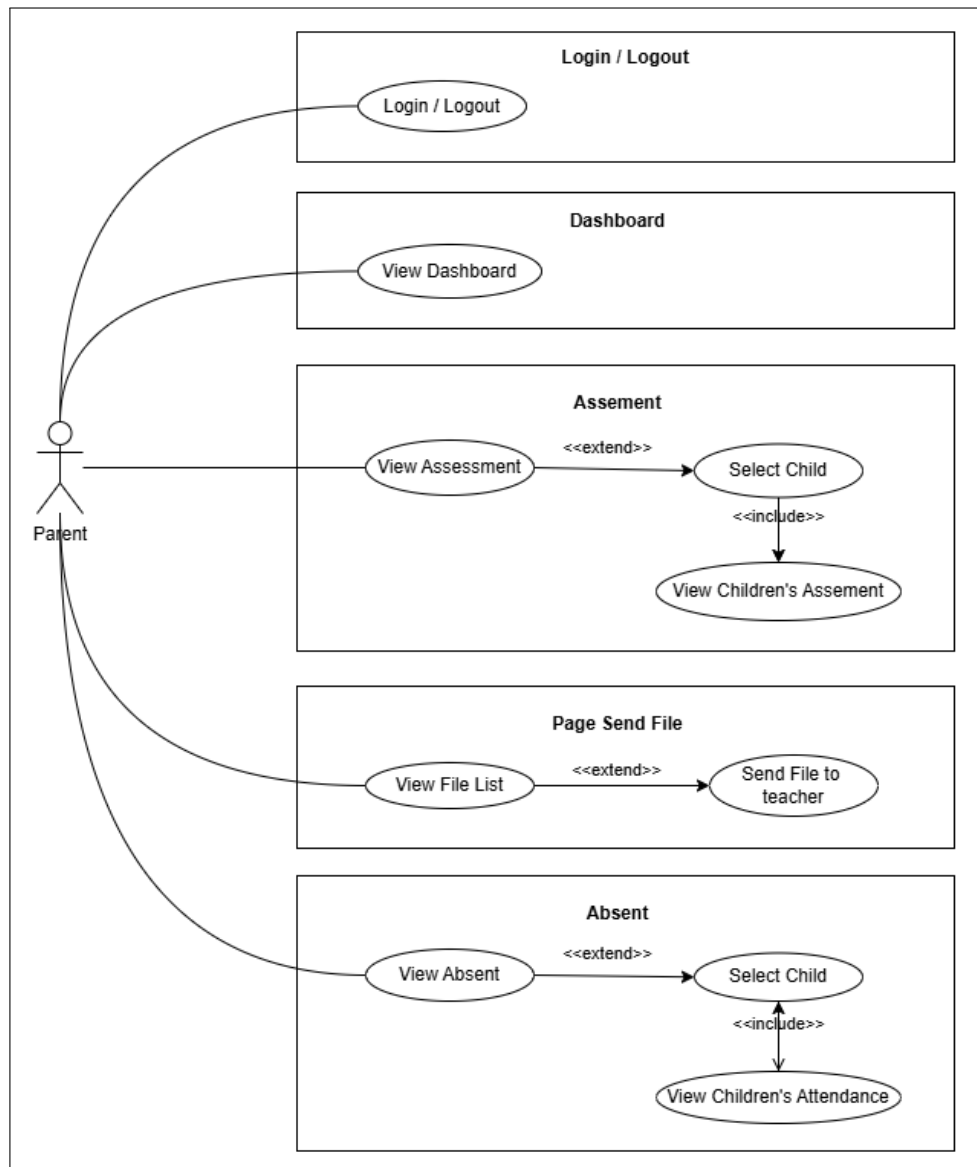
C Parent

Use Case Diagram Parent menggambarkan interaksi antara aktor *Parent* (orang tua) dengan sistem *Learning Management System (LMS)* , sebagaimana ditunjukkan pada Gambar 3.3. Orang tua memiliki akses terhadap beberapa fitur penting dalam sistem, yang difokuskan pada pemantauan perkembangan anak serta komunikasi dengan guru.

Interaksi dimulai dari proses *Login/Logout*, kemudian dilanjutkan dengan akses ke *Dashboard* sebagai halaman utama. Salah satu fitur utama bagi orang tua adalah *View Assessment*, yang digunakan untuk melihat penilaian anak. Fitur ini memiliki relasi *extend* dengan aksi *Select Child*, yang berfungsi untuk memilih anak yang ingin dilihat datanya. Setelah anak dipilih, sistem akan menampilkan asesmen masing-masing anak melalui relasi *include* ke fitur *View Children's Assessment*.

Selain itu, orang tua juga dapat menggunakan fitur *View File List* untuk mengakses daftar dokumen yang telah dikirim atau akan dikirim ke guru. Fitur ini diperluas dengan aksi *Send File to Teacher*, yang memungkinkan orang tua mengunggah file pendukung seperti tugas atau catatan kepada pihak bimbel atau guru yang bersangkutan. Diagram ini memperlihatkan bahwa peran orang tua dalam sistem tidak hanya sebatas pengamat, tetapi juga sebagai partisipan aktif dalam mendukung proses pembelajaran anak melalui pemantauan asesmen dan pengiriman file.





Gambar 3.3. Use Case Diagram Parent PT Matica Inovasi Edukasi

UNIVERSITAS
MULTIMEDIA
NUSANTARA

D Student

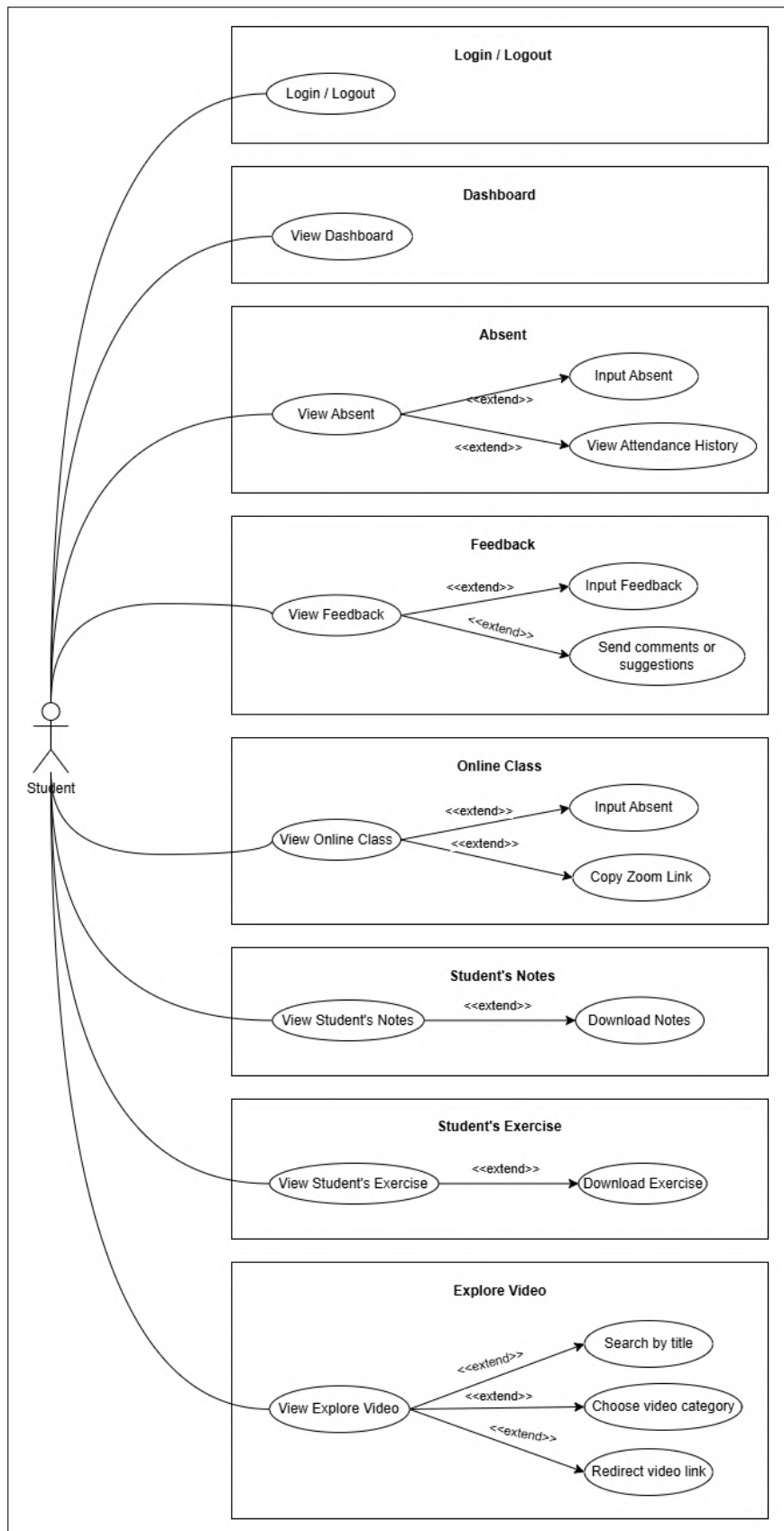
Use Case Diagram Student menggambarkan interaksi antara aktor *Student* (siswa) dengan *Learning Management System (LMS)* , seperti ditunjukkan pada Gambar 3.4. Siswa memiliki akses terhadap berbagai fitur pembelajaran yang mendukung kegiatan belajar secara daring maupun luring.

Siswa memulai interaksi dengan sistem melalui proses *Login/Logout*, kemudian diarahkan ke *Dashboard* sebagai pusat navigasi utama. Salah satu fitur penting adalah *View Absent*, yang memiliki relasi *extend* ke dua aktivitas, yaitu *Input Absent* dan *View Attendance History*. Hal ini memungkinkan siswa untuk mencatat kehadiran dan meninjau histori presensinya.

Selain itu, siswa juga dapat memberikan masukan melalui fitur *View Feedback* yang diperluas menjadi *Input Feedback* serta *Send Comments or Suggestions*. Untuk kebutuhan pembelajaran daring, siswa mengakses fitur *View Online Class* yang mencakup kemampuan untuk *Input Absent* secara otomatis saat mengikuti kelas dan menyalin *Zoom Link*.

Lebih lanjut, sistem menyediakan akses ke materi pembelajaran melalui fitur *View Student's Notes* dan *View Student's Exercise*, yang masing-masing dapat digunakan untuk *Download Notes* dan *Download Exercise*. Fitur *Explore Video* memungkinkan siswa menjelajahi konten video berdasarkan *Search by Title*, *Choose Video Category*, dan *Redirect Video Link*, yang terhubung sebagai ekstensi fitur utama. Dengan berbagai fitur tersebut, diagram ini menampilkan bagaimana siswa secara aktif mengakses, mengikuti, dan mengevaluasi proses pembelajaran melalui platform LMS yang disediakan.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.4. Use Case Diagram Student PT Matica Inovasi Edukasi

3.4.2 Sequence Diagram

Sequence Diagram adalah diagram dalam UML yang menggambarkan urutan interaksi antara objek atau komponen dalam sistem seiring berjalannya waktu. Diagram ini menunjukkan bagaimana objek saling berkomunikasi melalui pesan dalam urutan tertentu untuk menyelesaikan suatu fungsi atau proses. *Sequence Diagram* membantu dalam memodelkan interaksi dinamis antara objek dalam sistem, memberikan wawasan tentang bagaimana aliran kontrol dan data terjadi dalam konteks waktu. Diagram ini sangat berguna dalam merancang dan menganalisis interaksi antar objek dalam sistem [9].

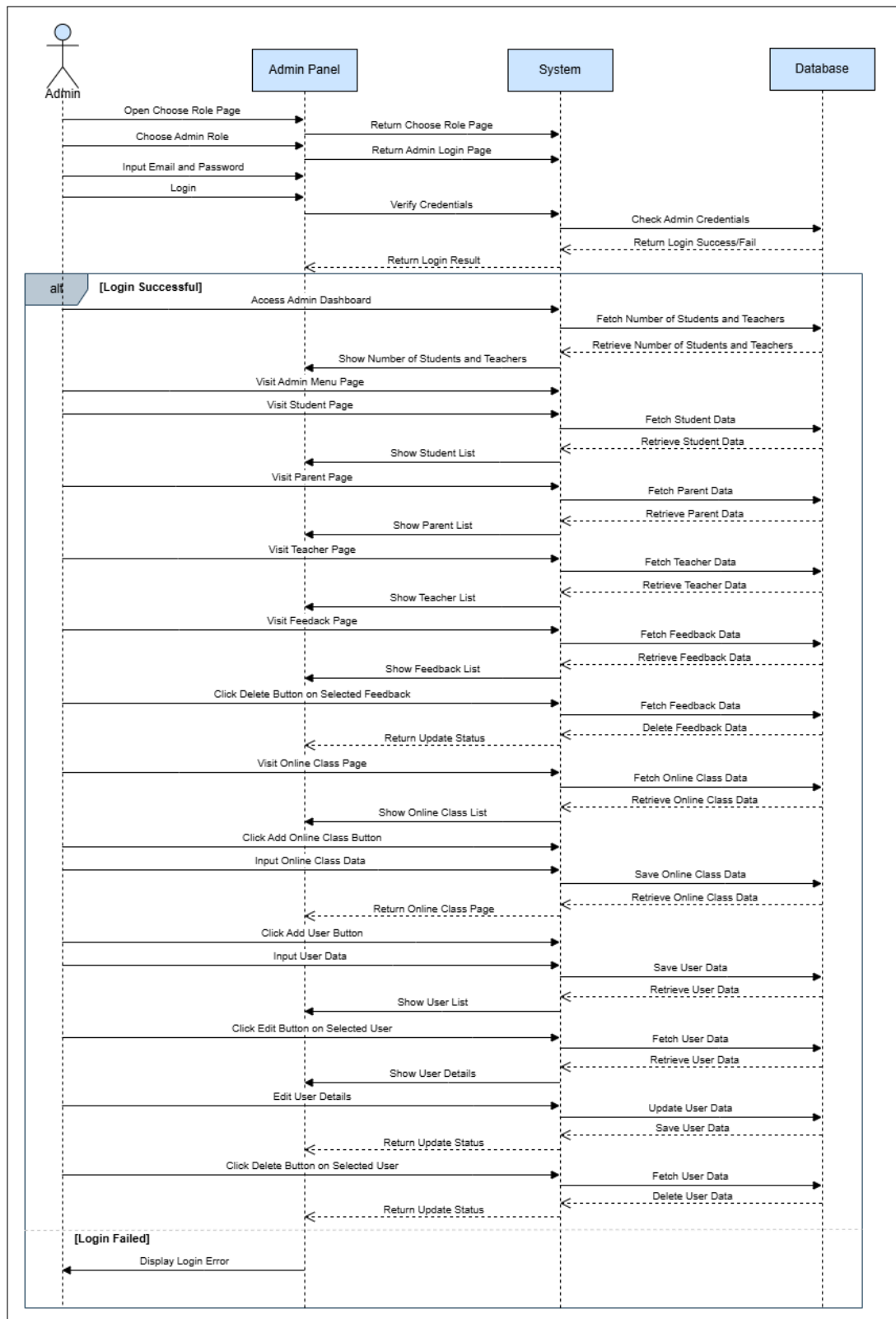
Dalam aplikasi LMS ini, *Sequence Diagram* digunakan untuk menggambarkan urutan interaksi antara pengguna, yaitu *Admin*, *Teacher*, *Parent*, dan *Student* dengan sistem selama proses tertentu, seperti login, presensi, pengiriman tugas, dan penilaian. Diagram ini menunjukkan langkah-langkah spesifik yang diambil oleh sistem dan pengguna, serta urutan pesan yang dipertukarkan antar objek dalam sistem untuk menyelesaikan fungsi tersebut. Dengan demikian, *Sequence Diagram* membantu dalam merancang dan memahami interaksi yang terjadi dalam sistem secara rinci dan terstruktur.

A Admin

Sequence Diagram Admin menggambarkan alur interaksi antara aktor *Admin* dan sistem pada platform manajemen data PT Matica Inovasi Edukasi, seperti pada Gambar 3.5. Proses dimulai saat *Admin* memilih peran dan diarahkan ke halaman *login* untuk memasukkan email dan kata sandi. Sistem memverifikasi kredensial tersebut melalui pencocokan data di *database*.

Ketika *login* berhasil, *Admin* masuk ke *dashboard* dan dapat mengakses halaman seperti *Student List*, *Parent List*, *Teacher List*, *feedback*, dan kelas *online*. Setiap akses halaman memicu sistem untuk mengambil data dari *database* dan menampilkannya di *Admin Panel*. *Admin* juga dapat menambah kelas *online*, mengelola pengguna, serta menghapus *feedback*. Tindakan ini akan memperbarui data di *database* dan mengembalikan status ke *Admin*.

Namun, jika proses *login* gagal, sistem akan menampilkan pesan kesalahan dan akses ke *dashboard* ditolak. Diagram ini memberikan gambaran menyeluruh tentang bagaimana interaksi antara *Admin*, sistem, dan *database* berlangsung dalam mendukung pengelolaan data internal platform.



Gambar 3.5. Sequence Diagram Admin PT Matica Inovasi Edukasi

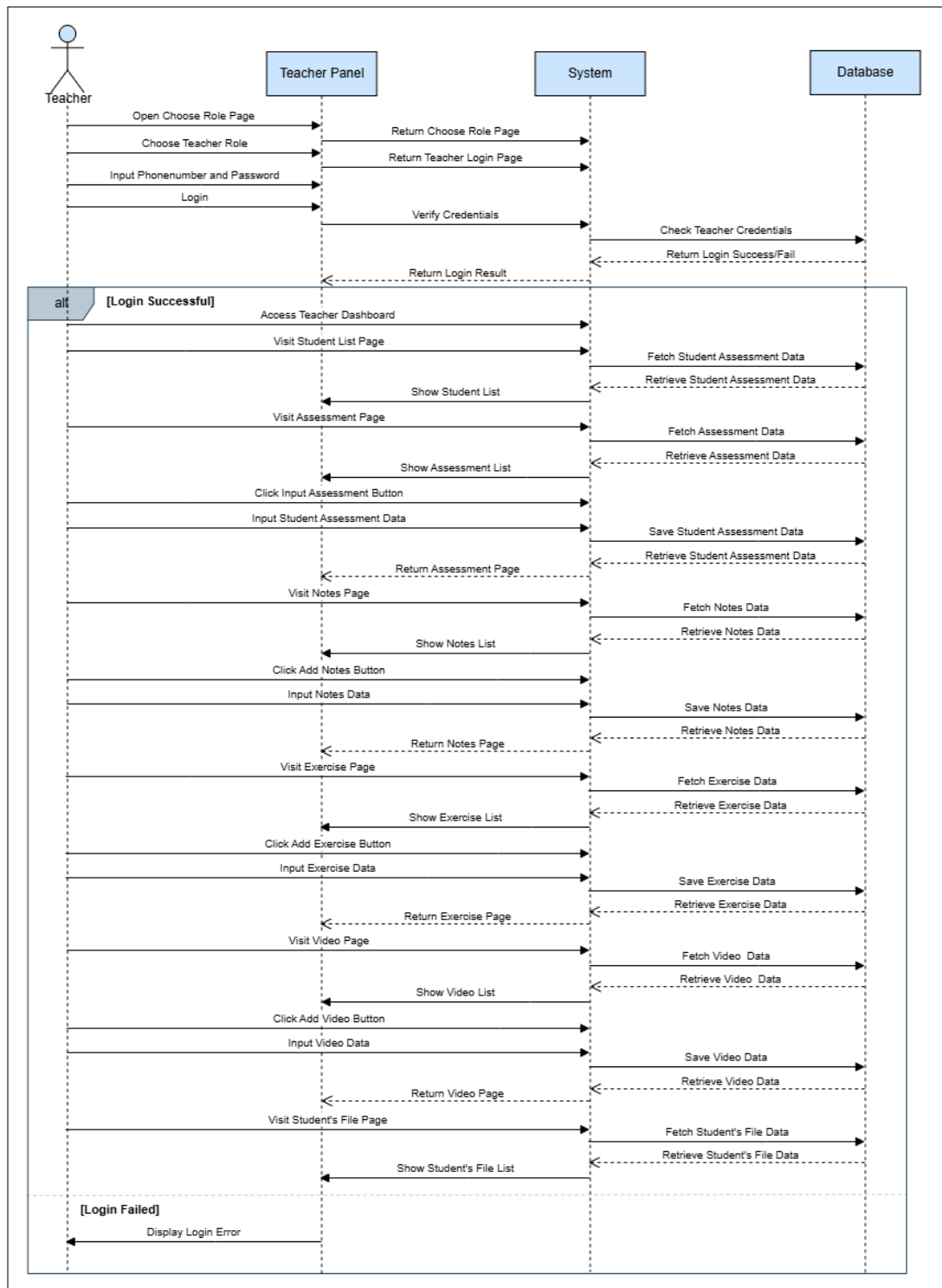
B Teacher

Sequence Diagram Teacher menjelaskan alur interaksi antara guru dan sistem pada platform manajemen pembelajaran milik PT Matica Inovasi Edukasi, sebagaimana ditampilkan pada Gambar 3.6. Proses diawali ketika *Teacher* mengakses halaman pemilihan peran dan memilih peran sebagai guru. Setelah diarahkan ke halaman *login*, *Teacher* memasukkan nomor telepon dan kata sandi sebagai kredensial *login*. Sistem kemudian memverifikasi data tersebut dengan mencocokkannya ke *database*. Jika *login* berhasil, *Teacher* diarahkan menuju *dashboard*.

Melalui *dashboard* ini, *Teacher* dapat mengakses berbagai halaman, seperti halaman daftar siswa, penilaian, catatan, latihan soal, video pembelajaran, dan file siswa. Pada setiap halaman yang diakses, sistem akan mengambil data dari *database* untuk ditampilkan dalam antarmuka *Teacher Panel*. *Teacher* dapat menambahkan penilaian siswa, membuat catatan, menambahkan latihan, serta mengunggah data video pembelajaran. Setiap aksi yang dilakukan akan disimpan dan diperbarui di *database* oleh sistem, dan hasilnya ditampilkan kembali kepada *Teacher*.

Diagram ini juga memperlihatkan bahwa *Teacher* dapat melihat daftar file siswa sebagai bagian dari pemantauan proses pembelajaran. Jika *login* gagal karena kesalahan kredensial, maka sistem akan menampilkan pesan kesalahan dan *Teacher* tidak akan dapat mengakses *dashboard*. Diagram ini memberikan gambaran komprehensif mengenai interaksi guru dengan sistem, serta alur data antara *Teacher Panel*, sistem inti, dan *database* dalam mendukung aktivitas pembelajaran digital.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.6. *Sequence Diagram Teacher PT Matica Inovasi Edukasi*

C Parent

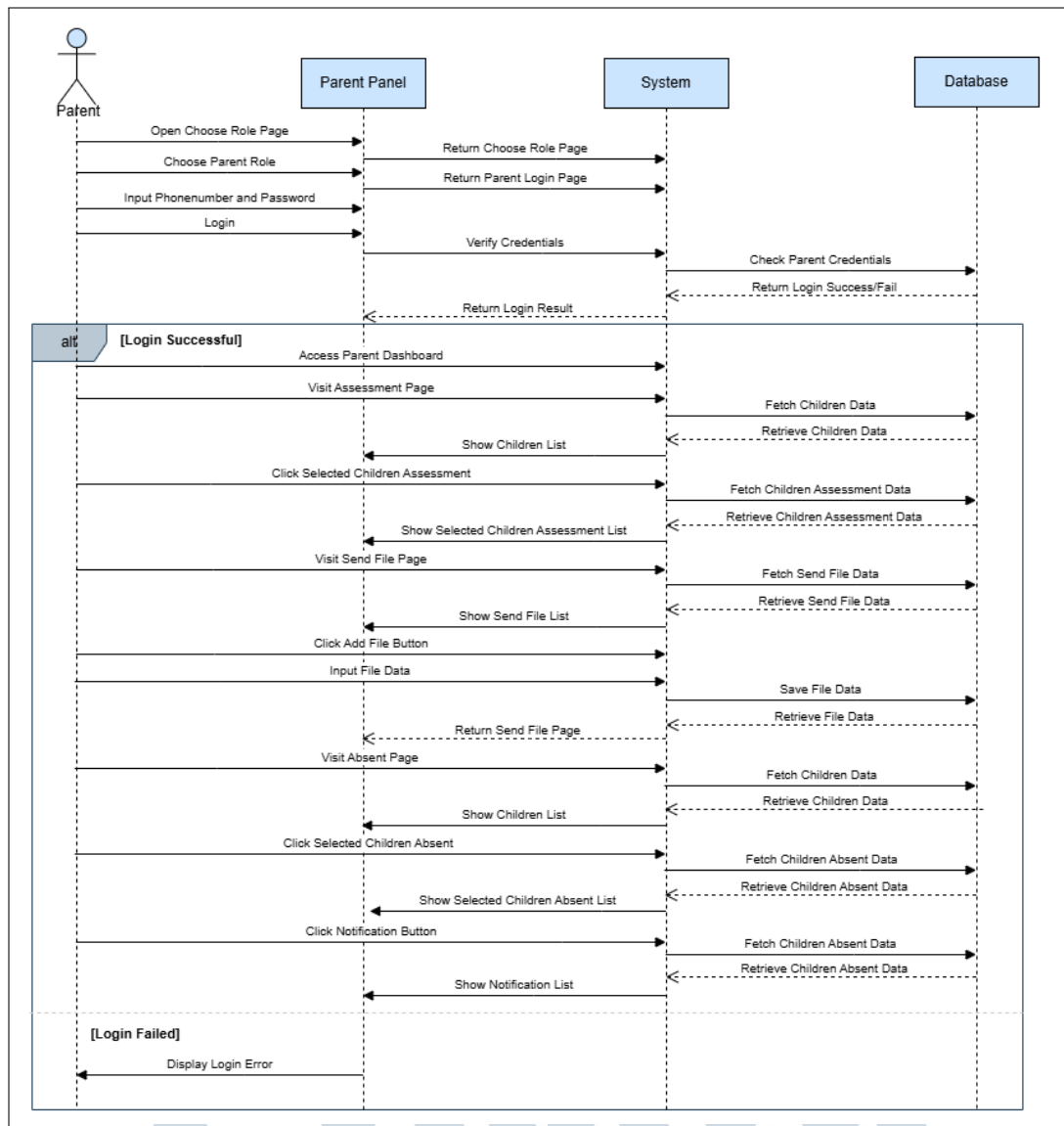
Sequence Diagram Parent menggambarkan alur interaksi antara orang tua (*Parent*) dan sistem pada platform digital milik PT Matica Inovasi Edukasi, sebagaimana ditunjukkan pada Gambar 3.7. Proses dimulai ketika *Parent* membuka halaman pemilihan peran dan memilih peran sebagai orang tua. Setelah diarahkan ke halaman *login*, *Parent* akan memasukkan nomor telepon dan kata sandi. Sistem kemudian memverifikasi kredensial tersebut dengan mencocokkannya ke dalam *database*.

Jika *login* berhasil, *Parent* akan diarahkan ke halaman *dashboard* dan dapat mengakses berbagai informasi terkait anak, seperti data penilaian, pengumpulan file, ketidakhadiran, serta notifikasi. Ketika mengakses halaman penilaian, *Parent* dapat memilih salah satu anak untuk melihat data penilaian secara detail. Selain itu, *Parent* juga dapat mengunggah file melalui halaman khusus dengan cara menambahkan data file yang kemudian disimpan oleh sistem ke dalam *database*.

Parent juga dapat mengakses halaman *attendance* untuk melihat catatan kehadiran anak, dan sistem akan menampilkan data yang relevan dari *database*. Fitur notifikasi juga tersedia untuk memberikan informasi terbaru mengenai status kehadiran anak. Apabila *login* gagal akibat kesalahan input, sistem akan menampilkan pesan kesalahan dan *Parent* tidak akan diarahkan ke *dashboard*.

Diagram ini memberikan visualisasi yang jelas tentang bagaimana peran orang tua terintegrasi dalam sistem, serta menggambarkan interaksi antara *Parent Panel*, sistem inti, dan *database* untuk mendukung pemantauan perkembangan dan aktivitas anak secara digital.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.7. Sequence Diagram Parent PT Matica Inovasi Edukasi

UNIVERSITAS
MULTIMEDIA
NUSANTARA

D Student

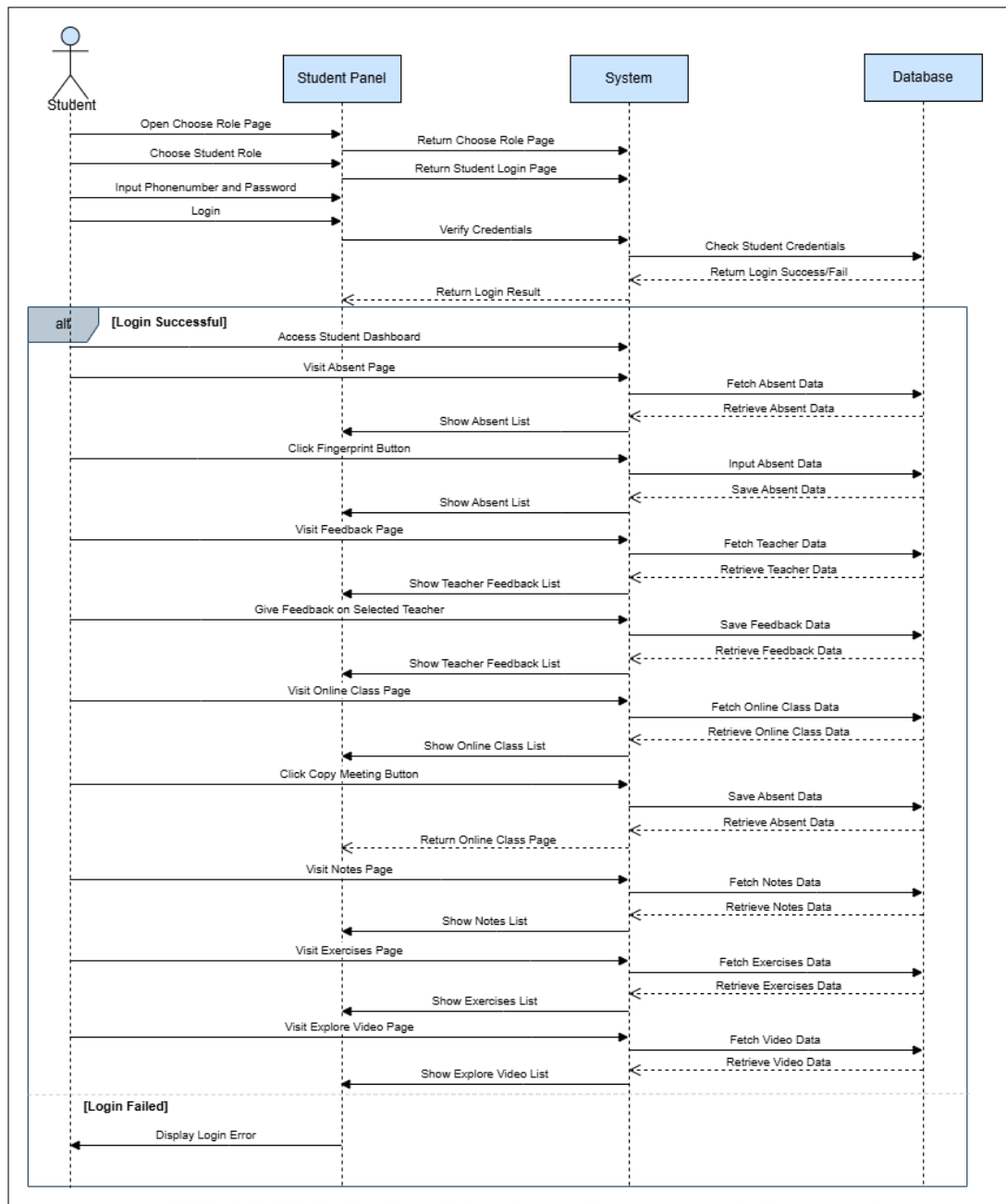
Sequence Diagram Student menggambarkan alur interaksi antara siswa dan sistem pembelajaran berbasis digital milik PT Matica Inovasi Edukasi, seperti yang diperlihatkan pada Gambar 3.8. Proses dimulai saat siswa membuka halaman pemilihan peran, memilih peran sebagai *Student*, lalu diarahkan ke halaman *login*. Setelah memasukkan nomor telepon dan kata sandi, sistem akan memverifikasi kredensial tersebut melalui data yang tersimpan dalam *database*.

Jika proses *login* berhasil, siswa akan diarahkan ke *dashboard* yang menyediakan berbagai fitur seperti melihat data kehadiran, memberikan *feedback* kepada guru, mengakses kelas *online*, catatan pembelajaran, latihan soal, dan video pembelajaran. Pada halaman kehadiran, siswa dapat melihat riwayat presensi serta melakukan input kehadiran menggunakan tombol sidik jari, yang datanya akan disimpan di dalam *database*.

Siswa juga memiliki akses untuk memberikan umpan balik terhadap guru yang datanya akan langsung tercatat oleh sistem. Selain itu, siswa dapat mengakses halaman kelas *online* dan menyalin tautan pertemuan, serta melihat catatan dan latihan yang telah tersedia. Sistem juga menyediakan halaman eksplorasi video, di mana siswa dapat melihat berbagai materi pembelajaran berbentuk video.

Apabila *login* gagal karena kredensial tidak sesuai, sistem akan menampilkan pesan kesalahan dan siswa tidak akan diarahkan ke *dashboard*. *Sequence Diagram* ini memberikan visualisasi lengkap mengenai peran siswa dalam sistem, serta menggambarkan integrasi antara *Student Panel*, sistem, dan *database* dalam mendukung aktivitas belajar secara digital.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.8. *Sequence Diagram Student* PT Matica Inovasi Edukasi

3.4.3 Activity Diagram

Activity Diagram merupakan salah satu diagram dalam notasi *Unified Modeling Language* (UML) yang berfungsi untuk menggambarkan alur kerja dari suatu proses, baik dari sudut pandang pengguna maupun sistem [8][9]. Diagram ini menyajikan urutan aktivitas dan keputusan yang dilakukan oleh aktor dalam

menjalankan fitur atau layanan yang tersedia pada sistem. Elemen-elemen penting dalam diagram ini antara lain adalah *initial node* sebagai titik awal, *action* untuk mewakili aktivitas, *decision* sebagai pengambilan keputusan logis, dan *final node* sebagai titik akhir [9].

Pada sistem *Learning Management System* (LMS) berbasis Android yang dikembangkan, terdapat empat aktor utama yaitu *Admin*, *Teacher*, *Parent*, dan *Student*. Setiap aktor memiliki alur aktivitas berbeda berdasarkan tanggung jawab dan hak akses yang dimiliki.

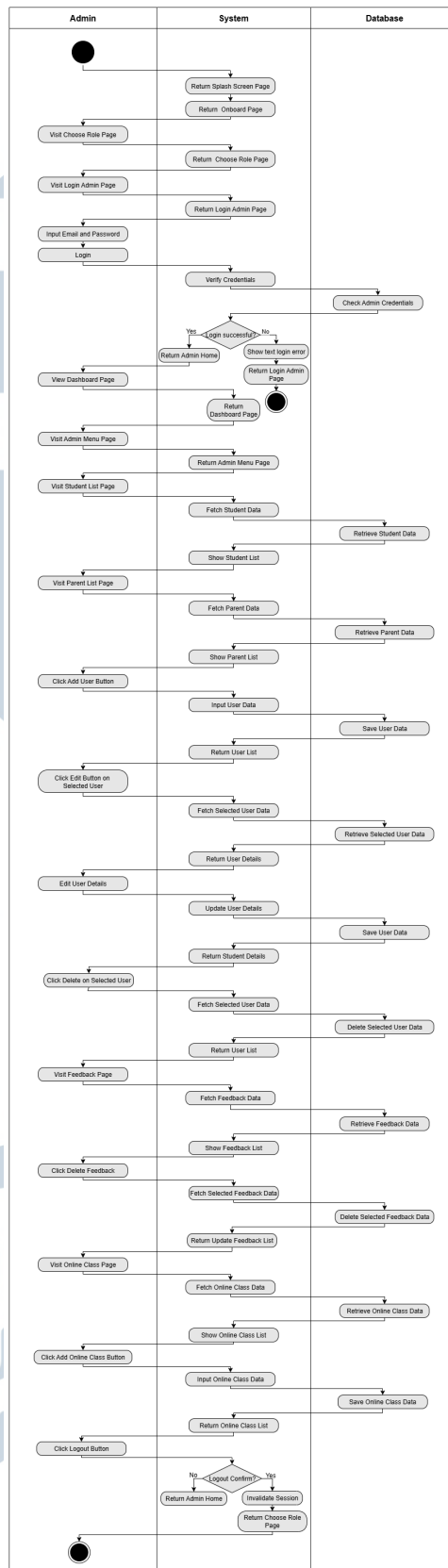
A Admin

Activity Diagram Admin menjelaskan alur aktivitas seorang admin dalam sistem *Learning Management System* (LMS) yang dikembangkan oleh PT Matica Inovasi Edukasi, seperti ditunjukkan pada Gambar 3.9. Admin bertanggung jawab dalam proses pengelolaan data pengguna dan kelas daring yang menjadi fondasi utama keberlangsungan sistem.

Alur dimulai ketika admin membuka aplikasi, memilih peran sebagai *Admin*, dan masuk melalui halaman *login*. Setelah berhasil masuk, admin diarahkan ke *dashboard* yang menyediakan berbagai fitur manajemen. Admin dapat menambahkan atau memperbarui data pengguna, baik itu siswa (*Student*), guru (*Teacher*), maupun orang tua (*Parent*) dengan mengakses fitur *Add User* atau *Update User*. Selain itu, admin juga memiliki akses untuk melihat dan mengelola daftar pengguna yang telah terdaftar dalam sistem.

Tidak hanya terbatas pada manajemen pengguna, admin juga dapat mengatur dan menjadwalkan kelas daring melalui fitur *Manage Zoom Class*. Fitur ini memungkinkan admin untuk menambahkan informasi kelas, jadwal, dan tautan pertemuan daring. Selain itu, admin juga dapat mengakses dan merespons *feedback* yang diberikan oleh siswa atau orang tua sebagai bagian dari evaluasi dan pengembangan sistem.

Proses ditutup ketika admin melakukan *logout* dari sistem. Diagram aktivitas ini memberikan gambaran yang sistematis mengenai tugas dan alur kerja admin dalam menjaga keberlangsungan data dan kelancaran operasional LMS.



Gambar 3.9. Activity Diagram Admin PT Matica Inovasi Edukasi

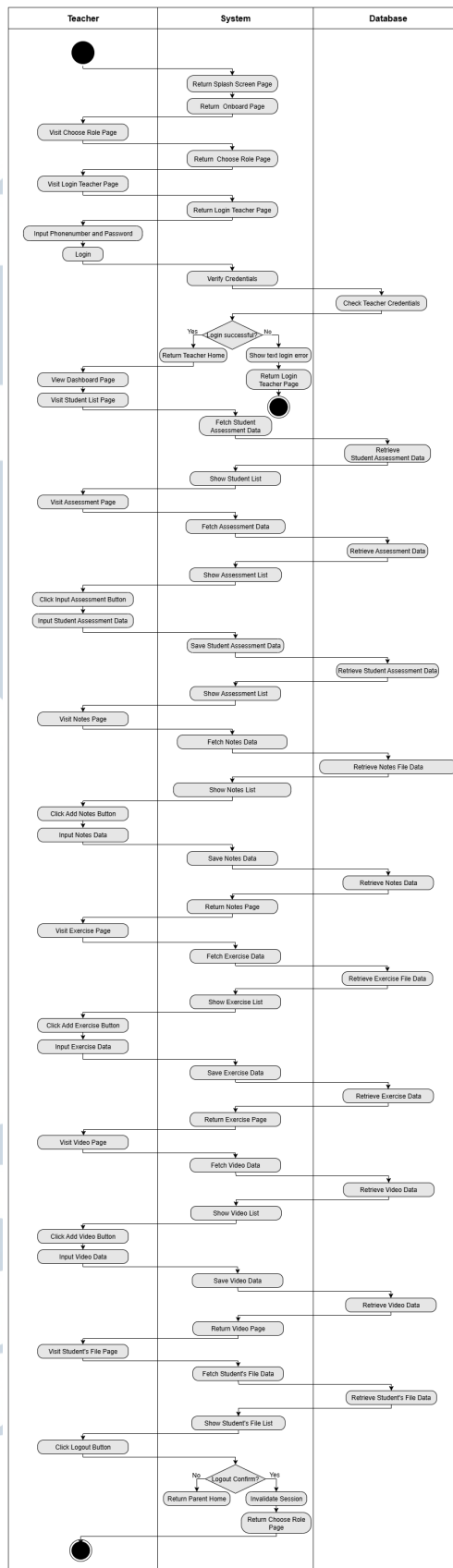
B Teacher

Activity Diagram Teacher pada Gambar 3.10 menunjukkan alur aktivitas pengguna dengan peran guru dalam sistem. Dimulai dari tampilan awal seperti *Splash Screen* dan *Onboard Page*, pengguna diarahkan ke halaman pemilihan peran, memilih *Teacher*, dan masuk ke halaman login. Setelah memasukkan nomor telepon dan kata sandi, sistem akan memverifikasi kredensial melalui data di *database*. Jika berhasil, pengguna diarahkan ke halaman utama guru.

Di halaman utama, guru dapat mengakses berbagai fitur seperti daftar siswa, asesmen, catatan, latihan, video pembelajaran, serta file dari siswa. Setiap halaman tersebut akan mengambil data dari *database* dan menampilkannya dalam bentuk daftar. Guru juga dapat menambahkan asesmen, catatan, latihan, atau video dengan mengisi data yang dibutuhkan dan menyimpannya ke sistem.

Saat mengakses halaman file siswa, guru dapat melihat daftar file yang telah dikirim oleh siswa atau orang tua. Semua data diambil dari *database* sesuai kebutuhan halaman. Jika guru ingin mengakhiri sesi, terdapat tombol *Logout* yang akan meminta konfirmasi sebelum mengembalikan pengguna ke halaman awal sistem.





Gambar 3.10. Activity Diagram Teacher PT Matica Inovasi Edukasi

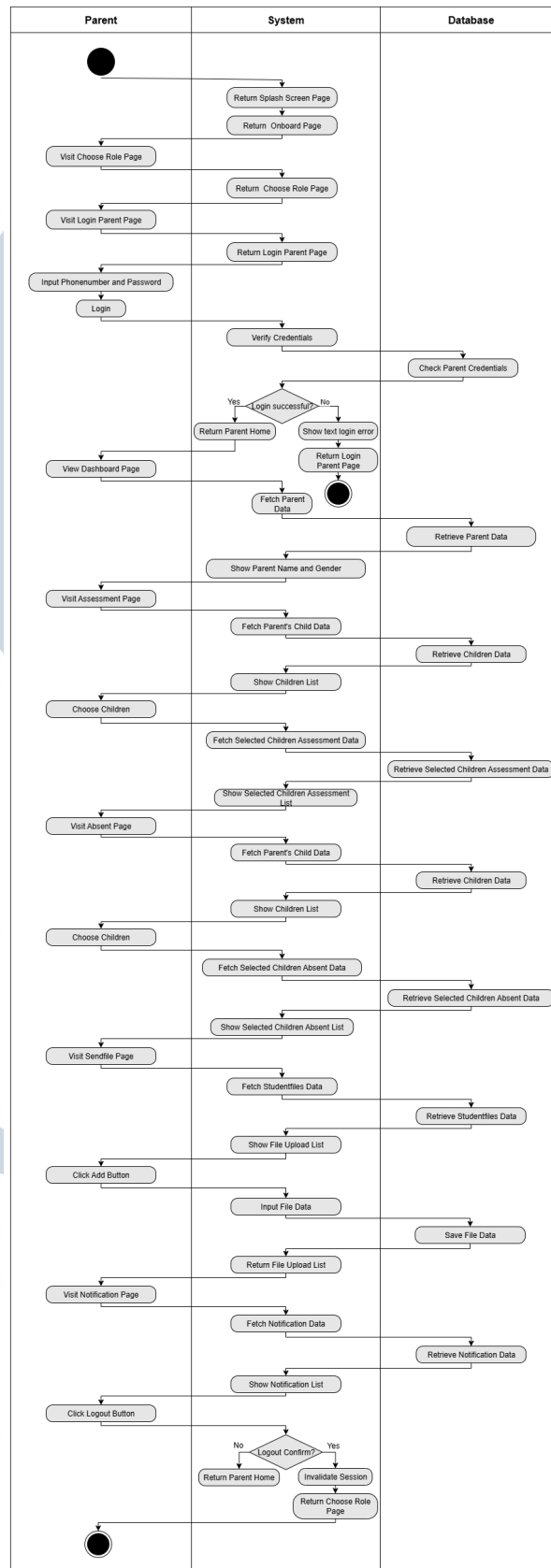
C Parent

Activity Diagram Parent menggambarkan alur aktivitas orang tua dalam menggunakan aplikasi. Seperti terlihat pada Gambar 3.11, proses diawali dari halaman *Splash Screen* dan *Onboard Page*, di mana pengguna diarahkan untuk memilih peran sebagai *Parent*. Selanjutnya, pengguna mengakses halaman login dan memasukkan nomor telepon serta kata sandi. Sistem kemudian memverifikasi kredensial dengan mencocokkannya pada *database*. Jika berhasil, pengguna diarahkan ke halaman utama *Parent*.

Di halaman utama, orang tua dapat mengakses fitur-fitur seperti halaman asesmen, presensi, pengiriman file, dan notifikasi. Pada halaman asesmen, orang tua dapat memilih anak yang ingin dilihat datanya. Data tersebut kemudian diambil dari *database* dan ditampilkan sebagai daftar asesmen. Proses serupa juga terjadi di halaman presensi, di mana daftar kehadiran anak ditampilkan berdasarkan pilihan pengguna.

Untuk pengiriman file, orang tua dapat mengunggah dokumen melalui halaman *Send File*. Data file ini ditampilkan dan dapat ditambahkan melalui tombol tertentu. Setiap file yang diunggah akan disimpan di penyimpanan sistem dan direkam informasinya di *database*. Selain itu, orang tua juga dapat melihat notifikasi yang ditampilkan berdasarkan data yang telah diambil dari *database*. Di akhir sesi, jika tombol *Logout* ditekan, sistem akan meminta konfirmasi dan mengembalikan pengguna ke halaman pemilihan peran setelah sesi dibatalkan.





Gambar 3.11. Activity Diagram Parent PT Matica Inovasi Edukasi

D Student

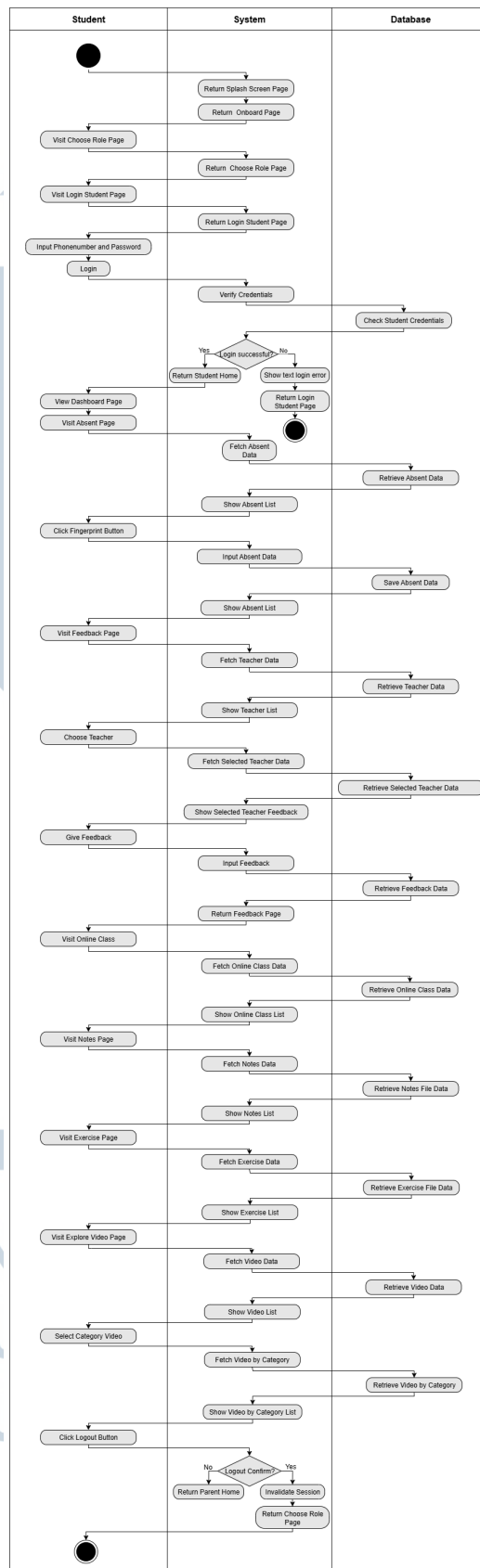
Activity Diagram Student menggambarkan alur aktivitas siswa saat menggunakan aplikasi. Proses dimulai dari tampilan awal seperti *Splash Screen* dan *Onboard Page*, kemudian siswa diarahkan ke halaman pemilihan peran dan login sebagai *Student*. Setelah memasukkan nomor telepon dan kata sandi, sistem melakukan verifikasi terhadap kredensial yang diberikan dengan mencocokkannya ke dalam *database*. Apabila data valid, pengguna diarahkan ke halaman utama.

Setelah berhasil masuk, siswa dapat mengakses berbagai fitur seperti presensi, pemberian *feedback*, kelas daring, catatan, latihan soal, dan eksplorasi video pembelajaran. Setiap fitur akan mengambil data terkait dari *database* dan menampilkannya dalam format daftar. Untuk fitur presensi, siswa dapat menekan tombol sidik jari untuk mencatat kehadiran yang kemudian disimpan secara otomatis.

Sementara itu, fitur *feedback* memungkinkan siswa memilih guru yang ingin diberi penilaian dan mengisi formulir umpan balik yang datanya tersimpan di sistem. Untuk kegiatan pembelajaran, siswa dapat mengikuti kelas daring, membaca catatan, mengerjakan latihan soal, dan melihat video pembelajaran yang dikategorikan. Daftar video dapat difilter berdasarkan kategori yang dipilih siswa.

Alur lengkap dari interaksi siswa dengan sistem dapat dilihat pada Gambar 3.12. Diagram tersebut juga menunjukkan proses *logout*, di mana siswa dapat keluar dari aplikasi melalui tombol *Logout*, dan sistem akan meminta konfirmasi sebelum mengakhiri sesi dan kembali ke halaman awal.

U M V N
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.12. Activity Diagram Student PT Matica Inovasi Edukasi

3.4.4 Class Diagram

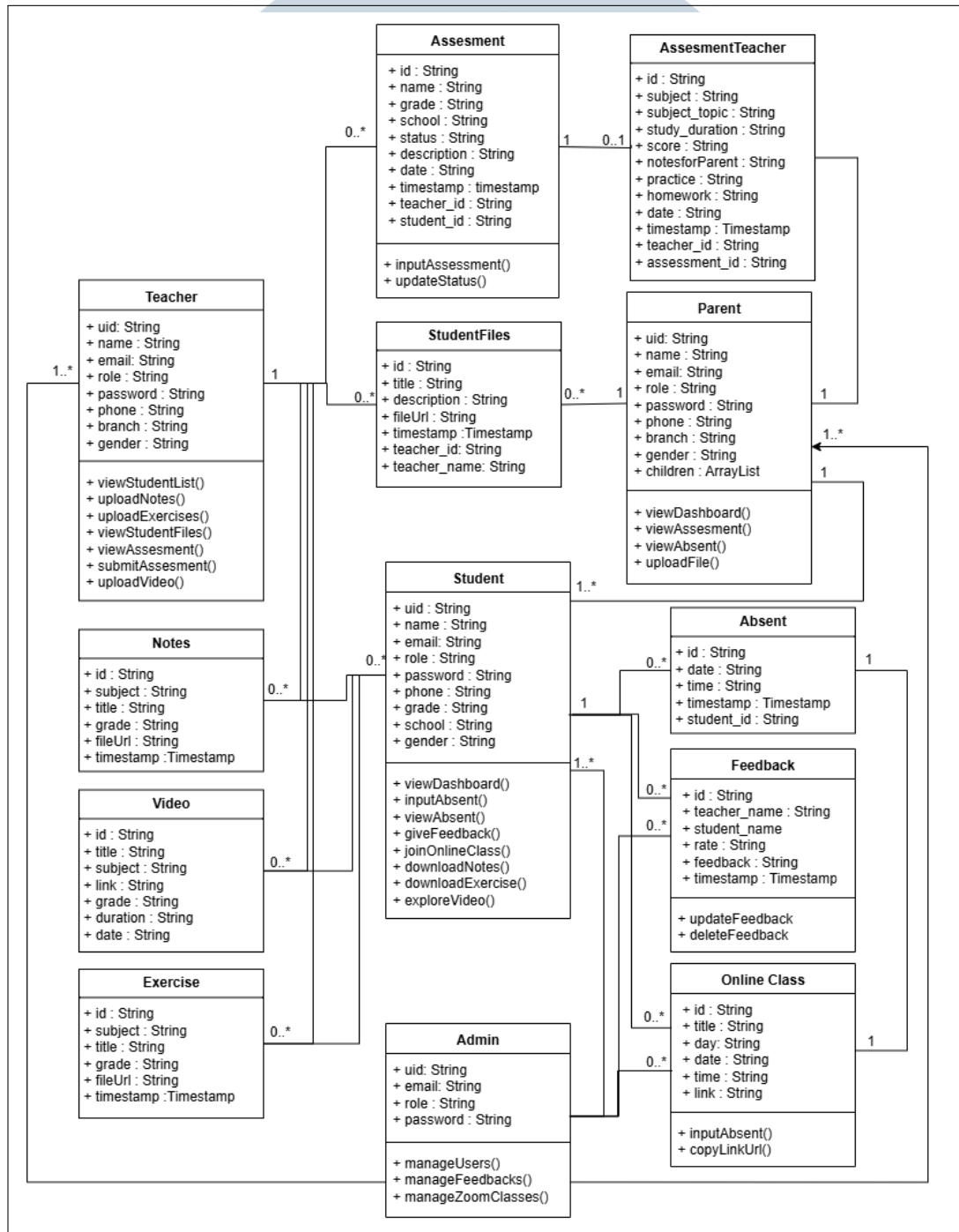
Class diagram merupakan salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang berfungsi untuk merepresentasikan struktur statis dari suatu sistem berbasis objek. Diagram ini menampilkan kelas-kelas dalam sistem, atribut, dan metode yang dimiliki oleh masing-masing kelas, serta hubungan antar kelas seperti asosiasi, agregasi, komposisi, dan generalisasi [8][9]. Selain itu, *class diagram* juga mencantumkan *multiplicity* atau kardinalitas yang menunjukkan jumlah keterhubungan antar objek, seperti satu-ke-banyak (*one-to-many*) atau banyak-ke-banyak (*many-to-many*). Dengan *class diagram*, pengembang perangkat lunak dapat memperoleh gambaran menyeluruh mengenai bagaimana data disusun dan bagaimana interaksi antar entitas dalam sistem dilakukan [9].

Class diagram yang ditampilkan pada Gambar 3.13 merepresentasikan struktur sistem *Learning Management System* (LMS) yang mencakup empat jenis pengguna utama, yaitu *Admin*, *Teacher*, *Student*, dan *Parent*. Masing-masing pengguna memiliki fungsi yang berbeda sesuai dengan perannya. *Admin* bertanggung jawab untuk mengelola pengguna lain, data *feedback*, dan kelas *online*, dengan metode seperti *manageUsers()*, *manageFeedbacks()*, dan *manageZoomClasses()*. *Teacher* dapat mengunggah materi pembelajaran seperti catatan, latihan, video, serta menilai siswa melalui kelas *Assessment* dan *StudentFiles*. *Student* memiliki akses terhadap materi pembelajaran, memberikan *feedback*, mengikuti kelas daring, serta melakukan presensi. Sementara itu, *Parent* dapat melihat perkembangan anak melalui *Assessment* dan *Absent*, serta mengunggah file yang berkaitan dengan siswa.

Dalam diagram ini, relasi antar kelas ditunjukkan menggunakan garis asosiasi yang disertai dengan *multiplicity* seperti 0..* dan 1..*, yang berarti satu objek bisa berelasi dengan nol atau lebih objek lainnya. Contohnya, setiap *Student* memiliki relasi *many-to-many* terhadap *Video*, *Exercise*, *Notes*, dan *Online Class*, menunjukkan bahwa siswa dapat mengakses banyak materi atau tidak sama sekali. *Parent* memiliki hubungan 1 1..* terhadap *Student*, yang berarti satu *Parent* harus memiliki minimal satu anak. *Teacher* memiliki hubungan dengan *Assessment* dan *StudentFiles*, karena guru bertanggung jawab atas penilaian dan pengelolaan file siswa. Di sisi lain, entitas seperti *Feedback*, *Absent*, dan *AssessmentTeacher* menjadi entitas pendukung yang merekam detail aktivitas pembelajaran siswa secara rinci.

Secara keseluruhan, *class diagram* ini memberikan representasi yang jelas

dan terstruktur mengenai hubungan antara pengguna dan entitas pembelajaran dalam sistem LMS. Dengan pemodelan ini, proses pengembangan sistem dapat dilakukan dengan lebih terarah dan efisien.



Gambar 3.13. *Class Diagram* PT Matica Inovasi Edukasi

3.4.5 Struktur Tabel Database

Struktur tabel pada sistem pembelajaran di PT Matica Inovasi Edukasi dirancang menggunakan Firebase *Firestore*, yang merupakan *database NoSQL* berbasis dokumen. Setiap data disimpan dalam bentuk *collection* dan *document*, dengan struktur yang fleksibel namun tetap mengikuti skema yang telah ditetapkan.

A Table teachers

Tabel 3.2 menjelaskan struktur tabel `teachers` yang digunakan untuk menyimpan informasi akun pengguna dengan peran sebagai guru dalam sistem. Data yang disimpan mencakup nama, cabang, alamat *email*, nomor telepon, *password*, peran pengguna, dan *uid*. Tabel ini digunakan untuk memfasilitasi proses login ke dalam halaman Guru pada aplikasi *Learning Management System* (LMS).

Tabel 3.2. Struktur Tabel `teachers`

Nama Atribut	Tipe Data
<code>uid</code>	string
<code>name</code>	string
<code>branch</code>	string
<code>email</code>	string
<code>phone</code>	string
<code>password</code>	string
<code>role</code>	string

Penjelasan Atribut Tabel `teachers`:

1. **`uid`**
Menyimpan kode unik untuk mengidentifikasi setiap guru dalam sistem.
2. **`name`**
Menyimpan nama lengkap dari guru yang terdaftar.
3. **`branch`**
Berisi informasi mengenai cabang tempat guru terdaftar.

4. **email**

Menyimpan alamat *email* yang digunakan oleh guru untuk login dan notifikasi.

5. **phone**

Menyimpan nomor telepon guru untuk kebutuhan komunikasi.

6. **password**

Digunakan untuk menyimpan kata sandi yang sudah di-*encrypt*.

7. **role**

Menandakan peran dari pengguna, dalam hal ini sebagai *teacher*.

B Parents

Tabel 3.3 menjelaskan struktur dari tabel *parents*, yang berfungsi untuk menyimpan data akun orang tua siswa dalam sistem. Data yang dicatat meliputi informasi dasar seperti nama, jenis kelamin, email, dan nomor kontak, serta cabang tempat pendaftaran dan daftar siswa yang merupakan anak dari orang tua tersebut. Tabel ini digunakan untuk menampilkan informasi akun orang tua pada tampilan aplikasi, dengan relasi langsung ke anak melalui atribut *children*. Setiap entri dalam tabel ini merepresentasikan satu akun orang tua yang dapat terhubung dengan satu atau lebih siswa.

Tabel 3.3. Struktur Tabel *parents*

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

Penjelasan Atribut Tabel *parents*:

1. **branch**

Menyimpan informasi mengenai cabang tempat siswa yang diasuh orangtua tersebut terdaftar.

2. **children**

Berisi daftar UID siswa yang berhubungan langsung dengan orangtua tersebut.

3. **email**

Mencatat alamat email yang digunakan oleh orangtua.

4. **gender**

Menyimpan informasi jenis kelamin orangtua, seperti male atau female.

5. **name**

Menyimpan nama lengkap orangtua.

6. **phone**

Menyimpan nomor telepon orangtua untuk keperluan kontak.

7. **role**

Menandakan peran pengguna sebagai orangtua dalam sistem (parent).

8. **uid**

Menyimpan kode identitas unik pengguna yang digunakan untuk autentikasi sistem.

C Students

Tabel 3.4 mendeskripsikan struktur tabel *students*, yang menyimpan data mengenai siswa yang terdaftar di dalam sistem. Informasi yang dicakup meliputi nama, jenis kelamin, cabang tempat belajar, informasi sekolah, jenjang kelas, dan identitas pengguna yang terhubung melalui sistem autentikasi. Tabel ini berguna untuk menampilkan daftar siswa kepada guru maupun orang tua, serta digunakan dalam proses akademik seperti presensi, penilaian, dan pemantauan perkembangan belajar siswa.

Tabel 3.4. 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**

Merupakan kode unik yang digunakan untuk mengidentifikasi setiap siswa secara individual.

2. **name**

Berisi nama lengkap siswa seperti yang terdaftar dalam sistem.

3. **gender**

Menyimpan jenis kelamin siswa, seperti *male* atau *female*.

4. **school**

Berisi informasi nama sekolah tempat siswa menempuh pendidikan formal.

5. **grade**

Menyimpan jenjang atau tingkat kelas siswa, seperti *Grade 1*, *Grade 2*, dan sebagainya.

6. **uid**

Menyimpan UID siswa yang digunakan untuk menghubungkan data siswa dengan akun pengguna dalam sistem.

D Exercise

Tabel 3.5 menggambarkan struktur penyimpanan data latihan soal yang diberikan oleh guru kepada siswa. Tabel ini mencakup informasi seperti judul latihan, deskripsi, topik mata pelajaran, jenjang kelas, serta tautan ke file soal. Fitur

ini memungkinkan guru untuk mengatur dan membagikan latihan secara digital, sementara siswa dapat mengakses latihan tersebut kapan saja melalui aplikasi. Atribut seperti `subject` dan `grade` membantu dalam pengkategorian materi sesuai kebutuhan.

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`**
Menandai identitas unik dari latihan soal yang disimpan.
2. **`title`**
Merupakan nama atau judul dari latihan soal.
3. **`subject`**
Menunjukkan mata pelajaran yang berhubungan dengan soal tersebut.
4. **`grade`**
Menyimpan level kelas yang sesuai dengan soal yang diberikan.
5. **`timestamp`**
Merekam waktu saat file latihan tersebut diunggah ke sistem.
6. **`file_url`**
Menyimpan tautan atau URL file soal yang dapat diakses atau diunduh oleh siswa.

E Notes

Tabel 3.6 mendokumentasikan struktur penyimpanan catatan atau materi pembelajaran yang dapat diakses oleh siswa. Informasi dalam tabel ini meliputi

judul materi, mata pelajaran terkait, kelas tujuan, waktu unggahan, serta tautan ke file materi. Guru dapat memanfaatkan tabel ini untuk mengunggah materi pembelajaran digital yang dapat diakses siswa secara mandiri melalui platform.

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**
Tautan yang merujuk pada file catatan atau materi pembelajaran.
2. **grade**
Menunjukkan jenjang pendidikan siswa yang menjadi sasaran materi.
3. **subject**
Menandai mata pelajaran yang berkaitan dengan catatan tersebut.
4. **timestamp**
Menyimpan waktu saat catatan diunggah ke dalam sistem.
5. **title**
Berisi judul catatan untuk mempermudah identifikasi kontennya.

F Student Files

Tabel 3.7 menyimpan informasi mengenai file yang dikirim oleh orang tua siswa kepada guru. Setiap entri memuat informasi berupa judul file, deskripsi isi file, identitas guru penerima (nama dan UID), serta waktu pengunggahan dan tautan file. Sistem ini membantu dalam memfasilitasi komunikasi formal antara orang tua dan guru, khususnya dalam pengiriman tugas, hasil kerja, atau dokumen penting secara langsung.

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`**
Menjelaskan isi atau maksud dari file yang dikirimkan.
2. **`fileUrl`**
Menyimpan link file yang dapat diakses oleh guru.
3. **`teacherName`**
Menyimpan nama guru tujuan yang menerima file dari orangtua.
4. **`teacherUid`**
UID guru yang digunakan untuk memastikan file hanya dapat diakses oleh guru yang bersangkutan.
5. **`timestamp`**
Menyimpan waktu saat file dikirim atau diunggah.
6. **`title`**
Menunjukkan judul file sebagai identifikasi utama.

G Videos

Tabel 3.8 digunakan untuk menyimpan konten pembelajaran berupa video yang dapat diakses oleh siswa. Setiap video disertai informasi seperti judul, mata pelajaran, jenjang kelas, durasi, tanggal unggah, dan tautan ke video tersebut. Video ini bertujuan mendukung pembelajaran mandiri siswa di luar sesi tatap muka, baik untuk penjelasan materi, latihan, maupun pembahasan soal.

Tabel 3.8. Struktur Tabel *videos*

Nama Atribut	Tipe Data
date	timestamp
duration	string
grade	string
link	string
subject	string
title	string

Penjelasan Atribut Tabel *videos*:

1. **date**
Menyimpan tanggal ketika video diunggah atau dipublikasikan.
2. **duration**
Mencatat durasi video pembelajaran.
3. **grade**
Menyimpan jenjang kelas target yang sesuai dengan video.
4. **link**
Tautan menuju video pembelajaran yang dapat diakses siswa.
5. **subject**
Menunjukkan bidang studi atau pelajaran yang dibahas dalam video.
6. **title**
Menyimpan judul video pembelajaran yang ditampilkan kepada siswa.

H Assessments

Tabel 3.9 menyimpan catatan terkait aktivitas belajar siswa, seperti kehadiran, status penilaian, dan laporan aktivitas pembelajaran. Informasi yang tercatat termasuk nama siswa, jenjang kelas, sekolah asal, waktu pencatatan, dan deskripsi laporan. Fungsinya adalah untuk merekam aktivitas harian siswa dan menyediakan data perkembangan belajar yang dapat dilihat oleh guru dan orang tua.

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 pencatatan kehadiran atau laporan siswa.
2. **description**
Menjelaskan aktivitas atau status penilaian siswa pada hari tertentu.
3. **grade**
Menyimpan Jenjang kelas siswa yang dinilai.
4. **name**
Menyimpan nama lengkap siswa yang dinilai.
5. **school**
Menyimpan nama sekolah tempat siswa terdaftar.
6. **status**
Menunjukkan status asesmen, misalnya apakah penilaian sudah dilakukan atau belum.
7. **timestamp**
Merekam waktu input laporan ke dalam sistem.
8. **uid**
UID unik dari siswa yang dilaporkan.

I Feedback

Tabel 3.10 menyimpan data mengenai umpan balik yang diberikan siswa kepada guru setelah sesi pembelajaran berlangsung. Informasi yang dicatat mencakup nama siswa dan guru, isi komentar atau masukan, penilaian dalam bentuk angka, serta waktu pemberian *feedback*. Tabel ini berguna untuk meningkatkan kualitas pembelajaran melalui evaluasi dari sudut pandang siswa.

Tabel 3.10. Struktur Tabel *feedback*

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

Penjelasan Atribut Tabel *feedback*:

1. ***feedback***

Komentar atau masukan dari siswa tentang pembelajaran atau guru.

2. ***rating***

Nilai evaluasi dalam bentuk angka, biasanya dalam skala tertentu (misalnya 1–5).

3. ***studentName***

Nama siswa yang memberikan penilaian.

4. ***teacherName***

Nama guru yang menerima umpan balik.

5. ***timestamp***

Menyimpan waktu saat umpan balik diberikan oleh siswa.

J Online Class

Tabel 3.11 menyimpan data terkait jadwal kelas online yang akan dilaksanakan. Setiap entri mencakup informasi tentang judul kelas, tanggal, hari, waktu pelaksanaan, dan tautan ke ruang kelas virtual. Tabel ini memungkinkan

siswa untuk melihat dan bergabung dalam sesi pembelajaran daring dengan mudah sesuai jadwal yang telah ditentukan.

Tabel 3.11. Struktur Tabel *onlineclass*

Nama Atribut	Tipe Data
date	string
day	string
link	string
time	string
title	string

Penjelasan Atribut Tabel *onlineclass*:

1. **date**
Menyimpan informasi tanggal kelas *online* akan dilaksanakan.
2. **day**
Menyimpan hari dalam format singkat, seperti Mon, Tue, Wed, dan seterusnya.
3. **link**
Menyimpan tautan menuju ruang kelas *online* (misalnya Zoom dan Google Meet) yang dapat diakses oleh siswa.
4. **time**
Menyimpan waktu pelaksanaan kelas dalam format jam.
5. **title**
Menunjukkan nama atau topik kelas daring yang dijadwalkan.

3.4.6 Hasil Implementasi

Hasil implementasi yang disajikan pada bagian ini merupakan bentuk realisasi dari proses rancang bangun sebagian fitur pada aplikasi *Learning Management System* (LMS) berbasis Android untuk PT Matica Inovasi Edukasi. Ruang lingkup implementasi meliputi halaman pemilihan peran (*select role*) serta antarmuka dan fungsionalitas untuk peran *Parent* dan *Student*. Setiap tampilan mencerminkan integrasi antara rancangan antarmuka pengguna, alur navigasi, serta proses pengambilan dan penyajian data yang telah dirancang pada

tahap sebelumnya. Dokumentasi ini bertujuan untuk menunjukkan ketercapaian implementasi terhadap kebutuhan sistem yang telah ditetapkan, sesuai dengan peran dan fungsi yang menjadi bagian dari pengembangan aplikasi LMS.

A Select Role

Tampilan awal aplikasi dirancang untuk memastikan bahwa setiap pengguna dapat mengakses fitur yang sesuai dengan perannya. Oleh karena itu, sebelum masuk ke dalam sistem, pengguna diminta untuk memilih peran terlebih dahulu, apakah sebagai *Student*, *Teacher*, *Parent*, atau *Admin*. Setiap pilihan disajikan dalam bentuk kartu yang dilengkapi ilustrasi agar mudah dikenali. Setelah memilih salah satu peran, pengguna dapat menekan tombol *Continue* untuk diarahkan ke halaman login yang sesuai. Pada Gambar 3.14, terlihat antarmuka pemilihan peran (*role*) sebelum proses *login* dilakukan.

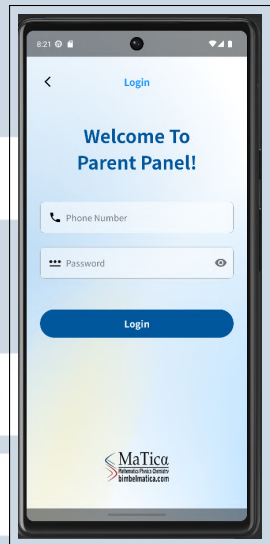


Gambar 3.14. Tampilan Halaman *Select Role*

B Login Parent

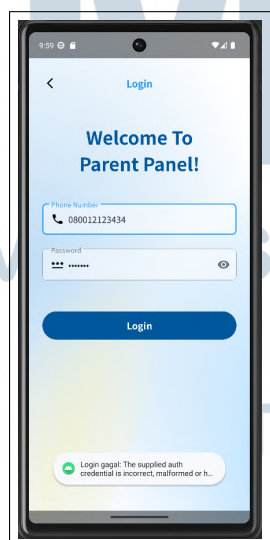
Setelah memilih peran sebagai *Parent*, pengguna akan diarahkan ke halaman *login* yang dikhususkan untuk peran tersebut. Pada halaman ini, pengguna diminta untuk memasukkan nomor telepon dan kata sandi yang telah terdaftar sebelumnya sebagai bentuk autentikasi. Proses ini memastikan bahwa hanya pengguna yang

memiliki kredensial yang sah yang dapat mengakses fitur-fitur khusus orang tua. Gambar 3.15 menampilkan tampilan halaman *login* khusus untuk peran *Parent*.



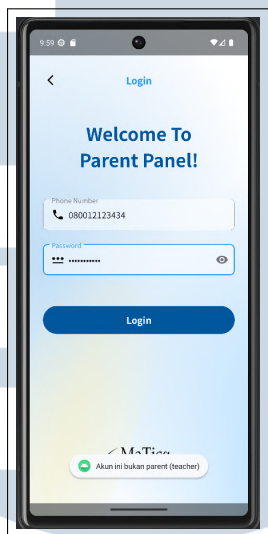
Gambar 3.15. Tampilan Halaman *Login Parent*

Apabila pengguna memasukkan nomor telepon atau kata sandi yang salah, sistem akan menampilkan *error message* yang menyatakan bahwa informasi *login* tidak valid. Hal ini bertujuan untuk memberi tahu pengguna bahwa terdapat kesalahan dalam memasukkan data akun. Gambar 3.16 menunjukkan tampilan ketika proses *login* gagal akibat kesalahan kredensial seperti nomor telepon atau *password* yang tidak sesuai.



Gambar 3.16. Tampilan Ketika Salah Kredensial

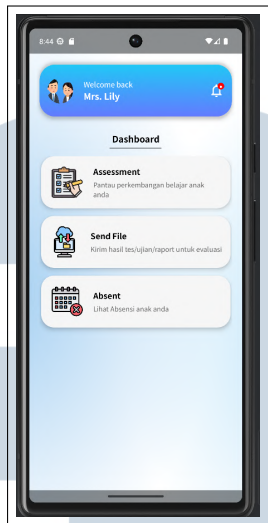
Sementara itu, jika pengguna telah memasukkan nomor telepon dan kata sandi dengan benar, tetapi memilih peran (*role*) yang salah saat *login*, misalnya memilih *Parent* padahal akun tersebut terdaftar sebagai *Teacher*. Maka akan muncul pesan kesalahan yang menginformasikan bahwa akun ini bukan *Parent*, melainkan *Teacher*. Pada Gambar 3.17, ditampilkan notifikasi kesalahan yang muncul apabila terjadi ketidaksesuaian antara peran yang dipilih dan data akun yang sebenarnya.



Gambar 3.17. Tampilan Ketika Salah Memilih *Role*

C Dashboard Parent

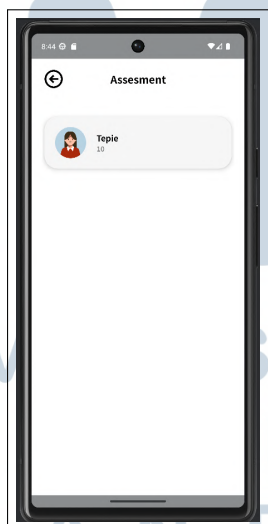
Setelah berhasil *login*, pengguna dengan peran *Parent* akan diarahkan ke halaman *dashboard* utama yang menyambut pengguna dengan ucapan selamat datang. Pada halaman ini, terdapat tiga fitur utama yang dapat diakses, yaitu *Assessment*, *Send File*, dan *Absent*. Masing-masing fitur dirancang untuk membantu orang tua dalam memantau perkembangan belajar dan kehadiran anak, serta mengirimkan dokumen terkait pembelajaran. Selain itu, sistem juga menyediakan fitur *notification* berupa penanda *badge* merah pada ikon notifikasi jika terdapat informasi kehadiran anak yang belum dibuka oleh orang tua. Gambar 3.18 menunjukkan tampilan *dashboard Parent*.



Gambar 3.18. Tampilan Awal Halaman *Parent*

D Assessment Parent

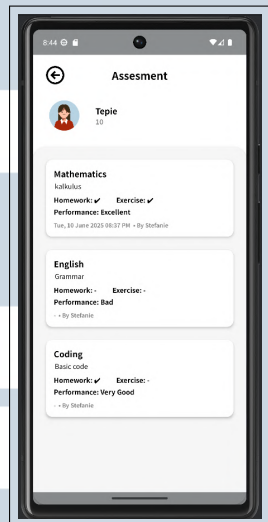
Untuk mengakses data penilaian, pengguna terlebih dahulu harus memilih anak yang ingin dilihat informasinya. Hal ini dilakukan agar sistem dapat menampilkan informasi yang spesifik sesuai dengan anak yang dimaksud, terutama bagi orang tua dengan lebih dari satu anak. Gambar 3.19 menampilkan tampilan daftar anak yang tersedia untuk dipilih sebelum mengakses detail *assessment*.



Gambar 3.19. Tampilan Halaman Memilih Anak Untuk Lihat *Assessment*

Setelah anak dipilih, sistem akan menampilkan daftar penilaian yang mencakup nama mata pelajaran, deskripsi materi, status *homework*, *exercise*, dan

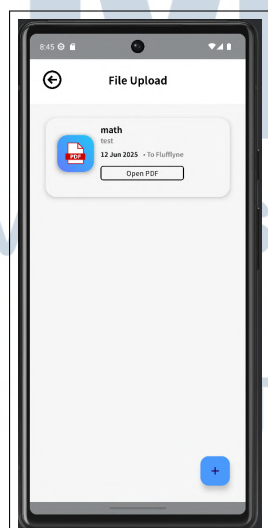
performance. Penilaian ini juga disertai dengan nama guru serta tanggal penilaian dilakukan. Gambar 3.20 memperlihatkan tampilan rincian penilaian (*assessment*) untuk seorang anak.



Gambar 3.20. Tampilan Halaman *Assesment*

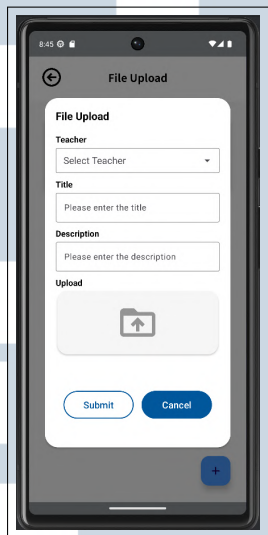
E Send File

Pada fitur *Send File*, orang tua dapat melihat daftar file yang telah dikirim kepada guru, seperti hasil tes, ujian, atau rapor. Setiap file disertai informasi berupa judul, deskripsi, tanggal pengiriman, dan nama guru tujuan. Tersedia juga tombol *download* file. Gambar 3.21 menunjukkan *list file* yang diunggah oleh orang tua.



Gambar 3.21. Tampilan Halaman *Send File*

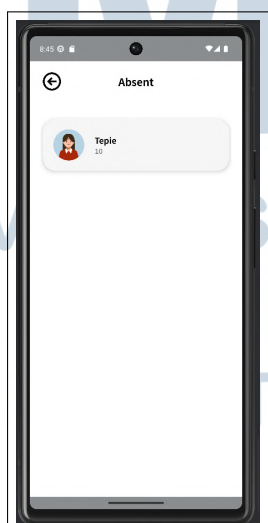
Untuk mengunggah file, orang tua dapat menekan tombol tambah (+) di kanan bawah, lalu mengisi formulir berisi pilihan guru, judul, deskripsi, dan unggahan dokumen. Setelah lengkap, file dikirim dengan menekan tombol *Submit*. Tampilan proses ini ditunjukkan pada Gambar 3.22.



Gambar 3.22. Tampilan Halaman *Add File*

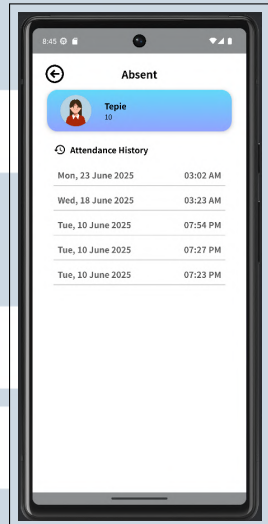
F Absent Parent

Sama seperti fitur *Assessment*, untuk melihat riwayat kehadiran anak, orang tua harus memilih terlebih dahulu anak yang ingin dilihat datanya. Gambar 3.23 memperlihatkan tampilan pemilihan anak sebelum melihat data presensi.



Gambar 3.23. Tampilan Halaman Memilih Anak Untuk Lihat *Absent*

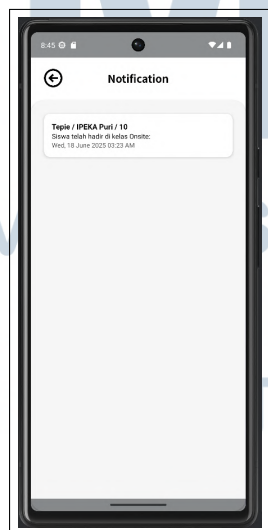
Setelah anak dipilih, sistem akan menampilkan riwayat kehadiran secara lengkap, yang mencakup tanggal dan waktu presensi. Gambar 3.24 memperlihatkan tampilan riwayat kehadiran anak yang telah dipilih.



Gambar 3.24. Tampilan Halaman *Absent Parent*

G Notification

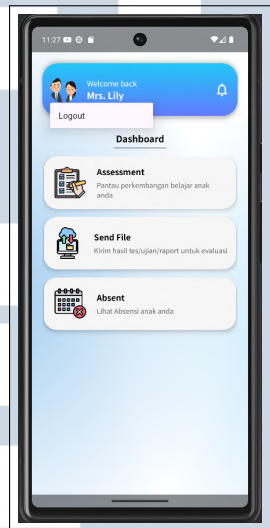
Setiap kali anak melakukan presensi baik secara *onsite* maupun *online class*, sistem akan mengirimkan notifikasi kepada orang tua yang berisi informasi nama anak, sekolah, kelas, serta waktu dan status kehadiran. Gambar 3.25 menunjukkan tampilan notifikasi kehadiran siswa yang diterima oleh orang tua.



Gambar 3.25. Tampilan Halaman *Notification*

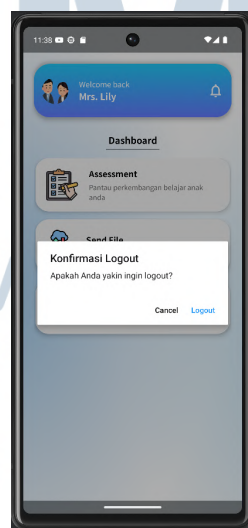
H Logout Parent

Orang tua dapat melakukan proses *logout* dengan menekan ikon profil di pojok kiri atas yang memunculkan opsi *Logout*. Gambar 3.26 menunjukkan tampilan menu *Logout* pada halaman utama akun orang tua.



Gambar 3.26. Tampilan menu *Logout* pada akun *Parent*

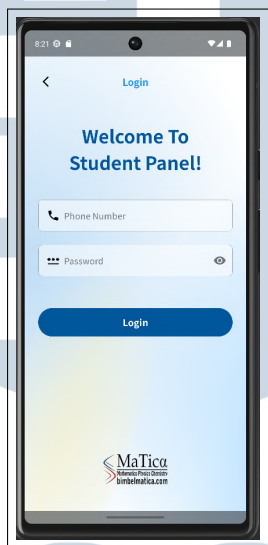
Setelah *logout* ditekan, sistem akan menampilkan dialog konfirmasi untuk memastikan apakah pengguna benar ingin keluar dari akun. Gambar 3.27 memperlihatkan tampilan dialog konfirmasi *logout* untuk akun orang tua.



Gambar 3.27. Dialog konfirmasi *logout* untuk *Parent*

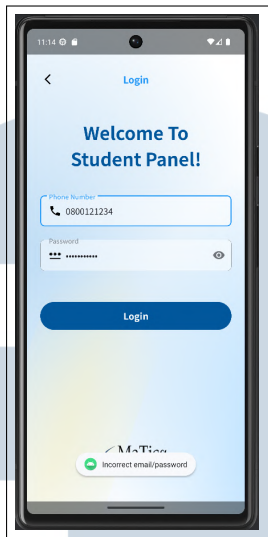
I Login Student

Pengguna yang memilih peran *Student* akan diarahkan ke halaman *login* khusus siswa. Pada halaman ini, siswa perlu memasukkan nomor telepon dan kata sandi yang telah didaftarkan sebelumnya untuk dapat mengakses *dashboard* pembelajaran. Proses ini memastikan bahwa hanya pengguna yang terverifikasi yang dapat menggunakan fitur-fitur yang tersedia dalam aplikasi. Gambar 3.28 menampilkan halaman *login* untuk siswa sebelum mengakses seluruh menu pembelajaran yang tersedia.



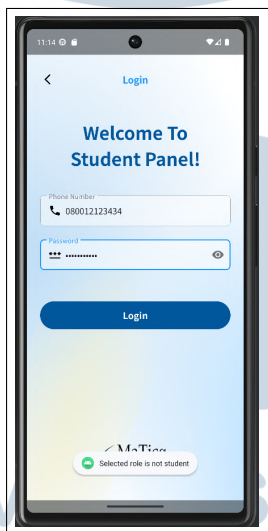
Gambar 3.28. Tampilan Halaman *Login Student*

Apabila pengguna memasukkan nomor telepon atau kata sandi yang salah, sistem akan menampilkan *error message* yang menyatakan bahwa informasi *login* tidak valid. Pesan ini muncul untuk memberi tahu pengguna bahwa kredensial yang dimasukkan tidak cocok dengan data yang tersimpan dalam sistem, sehingga proses autentikasi gagal dilakukan. Hal ini bertujuan menjaga keamanan akses terhadap akun pengguna. Gambar 3.29 menunjukkan tampilan ketika proses *login* gagal akibat kesalahan kredensial seperti nomor telepon atau *password* yang tidak sesuai.



Gambar 3.29. Tampilan Ketika Salah Kredensial

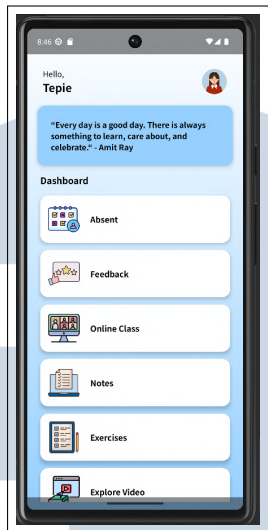
Sementara itu, jika pengguna telah memasukkan nomor telepon dan kata sandi dengan benar, tetapi memilih peran (*role*) yang salah saat *login*, maka akan muncul pesan kesalahan yang menginformasikan bahwa akun ini bukan *Student*. Gambar 3.30 menunjukkan pesan kesalahan saat peran yang dipilih salah.



Gambar 3.30. Tampilan Ketika Salah Memilih Role

J Dashboard Student

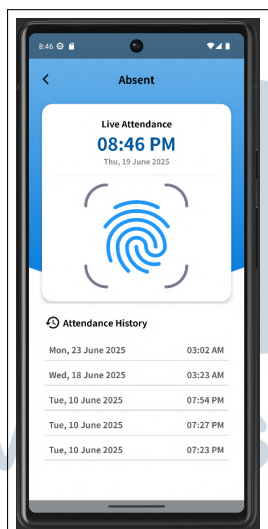
Setelah berhasil *login*, siswa akan masuk ke halaman *dashboard* yang berisi daftar fitur seperti *Absent*, *Feedback*, *Online Class*, *Notes*, *Exercises*, dan *Explore Video*. Gambar 3.31 menunjukkan tampilan *dashboard* siswa.



Gambar 3.31. Tampilan Awal Halaman *Parent*

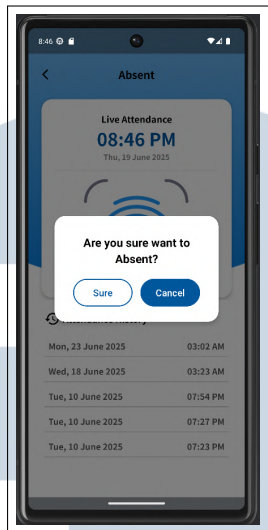
K AbsentStudent

Saat memilih fitur *Absent*, siswa akan diarahkan ke halaman *Absent*. Di bagian atas, terdapat penanda waktu dan tanggal saat ini, serta riwayat kehadiran. Gambar 3.32 memperlihatkan halaman *Absent Student*.



Gambar 3.32. Tampilan Awal Halaman *Absent Student*

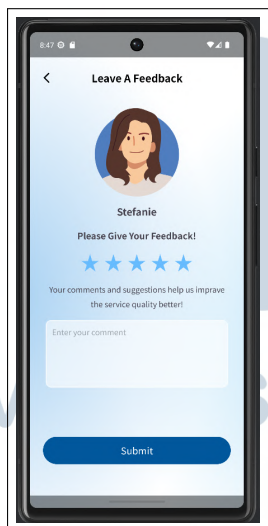
Sebelum proses presensi dicatat, akan muncul *popup* konfirmasi untuk memastikan siswa benar-benar ingin melakukan presensi. Gambar 3.33 menunjukkan tampilan *popup* konfirmasi yang harus dikonfirmasi sebelum lanjut.



Gambar 3.33. Tampilan Awal Halaman Konfirmasi *Absent Student*

L Feedback Teacher List

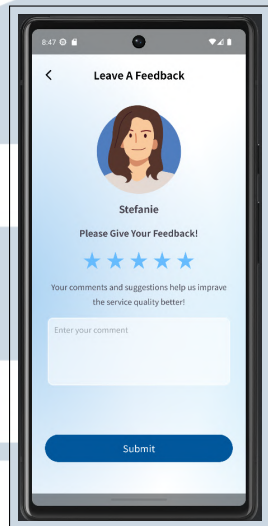
Fitur *Feedback* menampilkan daftar guru yang tersedia untuk diberikan penilaian atau umpan balik oleh siswa. Fitur pencarian juga disediakan untuk memudahkan pencarian guru. Gambar 3.34 menampilkan tampilan daftar guru dalam menu *Feedback* siswa.



Gambar 3.34. Tampilan Halaman *List Teacher*

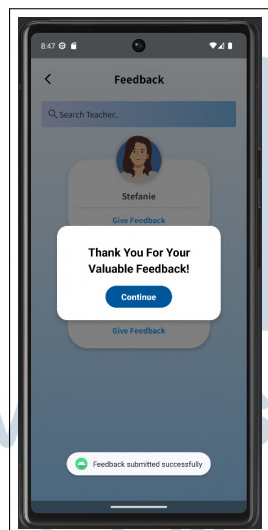
Setelah memilih guru dan menekan *Give Feedback*, siswa diarahkan ke halaman pengisian *feedback* dengan pilihan *rating* berupa bintang dan kolom

komentar. Gambar 3.35 memperlihatkan halaman pengisian *feedback* kepada guru oleh siswa.



Gambar 3.35. Tampilan Ketika Mengisi *Feedback*

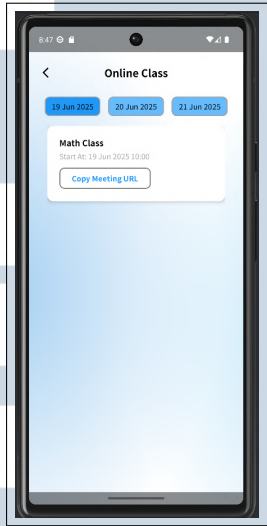
Setelah *feedback* dikirim, akan muncul notifikasi ucapan terima kasih atas masukan yang diberikan. Gambar 3.36 menunjukkan *popup* “*Thank You For Your Valuable Feedback*” sebagai tanda bahwa *feedback* berhasil dikirim.



Gambar 3.36. Tampilan Ketika Berhasil isi *Feedback*

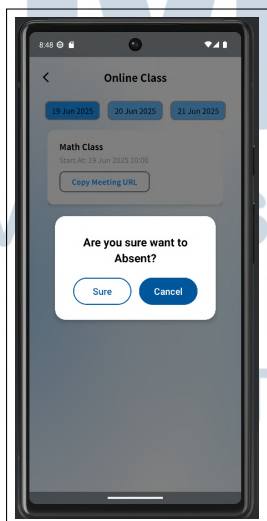
M Online Class

Fitur *Online Class* menampilkan daftar kelas *online* berdasarkan tanggal yang bisa dipilih siswa. Masing-masing kelas berisi informasi nama kelas dan waktu mulai. Gambar 3.37 memperlihatkan tampilan daftar *online class* siswa berdasarkan tanggal yang dipilih.



Gambar 3.37. Tampilan Halaman *Online Class*

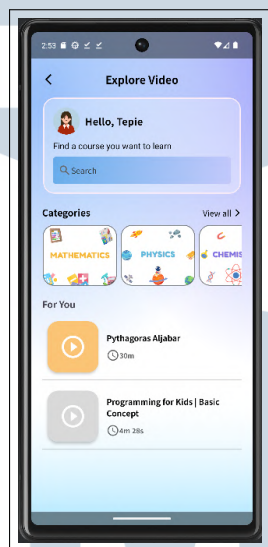
Ketika siswa menekan tombol "*Copy Meeting URL*", akan muncul *popup* konfirmasi apakah benar ingin mengikuti kelas sekaligus mencatat kehadiran. Gambar 3.38 menunjukkan tampilan *popup* konfirmasi sebelum bergabung ke kelas *online* dan melakukan presensi.



Gambar 3.38. Tampilan Halaman Konfirmasi *Absent Online Class*

N Explore Video

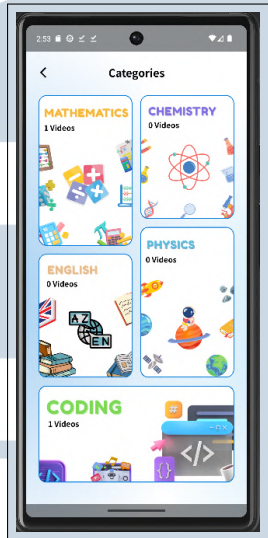
Pada menu *Explore Video*, siswa dapat menelusuri berbagai video pembelajaran yang tersedia di sistem. Halaman ini dirancang untuk memudahkan siswa dalam menemukan materi belajar secara mandiri sesuai kebutuhan masing-masing. Terdapat kolom *Search Video* yang memungkinkan pengguna mencari video berdasarkan kata kunci tertentu, serta bagian *For You* yang menampilkan rekomendasi video secara personal. Sistem dapat mengarahkan konten sesuai preferensi atau histori akses siswa. Selain itu, tombol *View All* disediakan agar siswa dapat melihat seluruh kategori video yang telah diklasifikasikan berdasarkan topik tertentu. Tampilan antarmuka pada halaman ini dirancang agar intuitif dan mudah digunakan, sehingga menunjang proses pembelajaran secara fleksibel dan mandiri. Gambar 3.39 memperlihatkan tampilan halaman *Explore Video* yang digunakan oleh siswa.



Gambar 3.39. Tampilan Halaman *Explore Video*

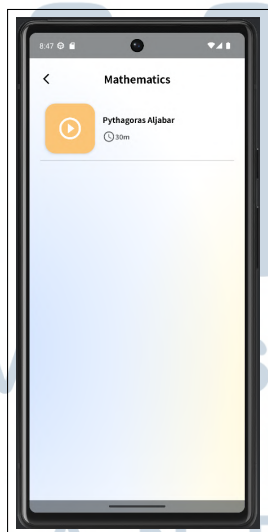
Saat menekan tombol *View All*, siswa akan diarahkan ke halaman kategori video pembelajaran yang telah dikelompokkan berdasarkan mata pelajaran. Halaman ini menyajikan daftar kategori seperti *Mathematics*, *Chemistry*, *Physics*, *English*, dan *Coding*, yang masing-masing dilengkapi dengan jumlah video yang tersedia di dalamnya. Dengan adanya pembagian kategori ini, siswa dapat menelusuri materi pembelajaran secara lebih terstruktur dan fokus sesuai dengan bidang studi yang diinginkan. Fitur ini juga membantu meningkatkan efisiensi pencarian serta memungkinkan eksplorasi konten yang relevan. Gambar 3.40

menampilkan tampilan daftar kategori video pembelajaran yang tersedia untuk siswa.



Gambar 3.40. Tampilan Halaman *Subject Category*

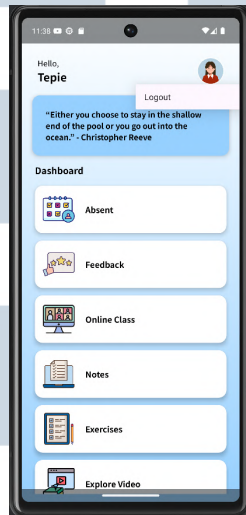
Jika siswa memilih salah satu kategori yang tersedia, seperti *Mathematics*, maka akan ditampilkan video-video yang tersedia dalam kategori tersebut. Gambar 3.41 memperlihatkan tampilan daftar video berdasarkan kategori yang dipilih oleh siswa.



Gambar 3.41. Tampilan Halaman Video Sesuai *Category*

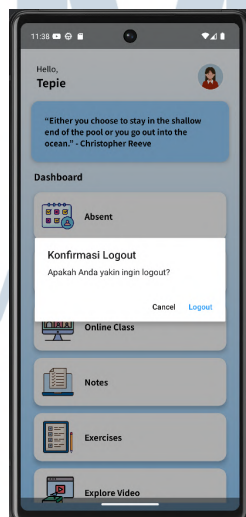
O Logout Student

Tampilan menu *Logout* untuk akun siswa dapat diakses melalui ikon profil di pojok kiri atas yang memunculkan opsi *Logout*. Gambar 3.42 menunjukkan tampilan opsi logout yang tersedia bagi siswa.



Gambar 3.42. Tampilan menu *Logout* pada akun Student

Saat tombol *Logout* ditekan, sistem akan menampilkan kotak dialog untuk mengonfirmasi tindakan keluar dari akun. Gambar 3.43 memperlihatkan tampilan dialog konfirmasi *logout* yang muncul pada akun siswa.



Gambar 3.43. Dialog konfirmasi *logout* untuk Student

3.5 Pengujian Sistem

Pengujian sistem dilaksanakan memakai pendekatan *Black Box testing* guna memastikan setiap fungsi utama aplikasi mulai dari autentikasi pengguna hingga manajemen konten pembelajaran dapat berjalan sesuai spesifikasi tanpa menelusuri kode sumber. Penguji hanya berfokus pada interaksi *input* dan *output* yang terlihat di antarmuka, mencakup skenario seperti autentikasi (*login/logout*), unggah berkas, penayangan video, hingga pencatatan asesmen. Seluruh skenario dijalankan dalam kondisi normal maupun tepi, misalnya memasukkan kredensial tidak valid atau mengunggah berkas dengan ukuran melebihi batas. Pengujian sistem terkait dilakukan oleh *Supervisor*, sehingga setiap anomali dapat segera dicatat dan direproduksi. Tabel 3.12 merangkum hasil pengujian untuk tiap fitur, meliputi setiap skenario yang diuji, hasil yang diharapkan, hasil pengujian aktual, serta kesimpulan dari setiap pengujian yang dilakukan.

Tabel 3.12. Hasil Pengujian Sistem Menggunakan Metode Black-Box

No.	Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Memeriksa pemilihan <i>role Parent</i>	<i>Parent</i> berhasil menuju ke halaman <i>Login Parent</i>	Dapat Menampilkan Halaman <i>Login Parent</i>	Sesuai
2	Memeriksa pemilihan <i>role Student</i>	<i>Student</i> berhasil menuju ke halaman <i>Login Student</i>	Dapat Menampilkan Halaman <i>Login Student</i>	Sesuai
3	Memeriksa <i>Login</i> dengan kredensial valid	Pengguna berhasil masuk ke <i>dashboard</i>	Pengguna dapat masuk tanpa kendala	Sesuai
4	<i>Login</i> dengan kredensial tidak valid	Sistem menolak dan menampilkan pesan kesalahan	Sistem menolak dan menampilkan “ <i>Incorrect Email/Password</i> ”	Sesuai
Lanjut pada halaman berikutnya				

Tabel 3.12 Hasil Pengujian Sistem Menggunakan Metode Black-Box (lanjutan)

No.	Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
5	<i>Login</i> dengan pemilihan <i>role</i> yang salah	Sistem menolak dan menampilkan pesan kesalahan	Sistem menolak dan menampilkan " <i>Selected role is not Student</i> "	Sesuai
6	Memeriksa tampilan <i>Dashboard Parent</i> setelah login berhasil	<i>Dashboard</i> menampilkan informasi <i>Parent</i> yang login	<i>Dashboard</i> berhasil menampilkan nama dengan imbuhan <i>Mrs.</i> sesuai <i>gender</i>	Sesuai
7	Memeriksa tampilan <i>Assessment</i> pada <i>Parent</i>	Menampilkan data anak yang ingin dilihat <i>assessment</i> nya	Menampilkan anak bernama " <i>Tepie</i> " dan <i>Grade</i> " <i>10</i> ".	Sesuai
8	Memeriksa tampilan <i>Assessment</i> anak yang dipilih pada <i>Parent</i>	Menampilkan data asses anak yang dipilih	Menampilkan <i>list assessment</i> yang telah diinput oleh guru.	Sesuai
9	Memeriksa tampilan <i>studentfiles</i> pada <i>Parent</i>	Berkas yang pernah diunggah tampil dan tautan tampil di daftar guru tujuan	File PDF berhasil diunggah dan terlihat di akun guru	Sesuai
10	Memeriksa tampilan <i>Absent</i> pada <i>Parent</i>	Menampilkan data anak yang ingin dilihat absennya	Menampilkan data anak bernama " <i>Tepie</i> " dan <i>Grade</i> " <i>10</i> ".	Sesuai
11	Memeriksa tampilan <i>Absent</i> anak yang dipilih pada <i>Parent</i>	Menampilkan data <i>absent</i> anak yang dipilih	Menampilkan <i>list attendance history</i> " <i>Tepie</i> ".	Sesuai
Lanjut pada halaman berikutnya				

Tabel 3.12 Hasil Pengujian Sistem Menggunakan Metode Black-Box (lanjutan)

No.	Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
12	Memeriksa <i>Badge Notification</i>	Sistem menampilkan <i>badge</i> merah jika halaman <i>notification</i> belum dibuka	<i>Badge</i> merah muncul	Sesuai
13	Memeriksa tampilan <i>Notification</i>	<i>Notification</i> setiap anaknya melakukan presensi akan muncul data berupa hari, tanggal, jam, dan status hadir	<i>Notification</i> berisi data "Wed, 18 June 2025 16:00" dan hadir kelas <i>Onsite</i> .	Sesuai
14	Memeriksa tampilan <i>Dashboard Student</i> setelah login berhasil	<i>Dashboard</i> menampilkan informasi <i>Student</i> yang login	<i>Dashboard</i> berhasil menampilkan nama dengan <i>profile</i> sesuai <i>gender</i>	Sesuai
15	Memeriksa tampilan <i>Absent</i> pada <i>Student</i>	Menampilkan <i>Attendance History</i>	Halaman <i>Absent</i> menampilkan data <i>Attendance History Student</i> .	Sesuai
16	Memeriksa konfirmasi <i>Absent</i> pada <i>Student</i>	Konfirmasi <i>Absent</i> muncul sebelum terdata presensinya	Setelah menekan tombol <i>fingerprint</i> maka konfirmasi presensi muncul untuk memverifikasi presensi.	Sesuai
17	Memeriksa <i>search bar</i> halaman <i>feedback</i> bekerja	Dapat mencari nama <i>Teacher</i>	Nama <i>Teacher</i> yang muncul sesuai dengan pencarian	Sesuai
Lanjut pada halaman berikutnya				

Tabel 3.12 Hasil Pengujian Sistem Menggunakan Metode Black-Box (lanjutan)

No.	Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
18	Memeriksa halaman <i>give feedback</i>	Menampilkan data <i>Teacher</i> yang akan diberi <i>feedback</i> dan memunculkan <i>popup Thankyou</i>	Data <i>Teacher</i> sesuai dan dapat menginput <i>feedback</i> dengan mengeluarkan <i>popup "Thankyou For Your Valuable Feedback"</i>	Sesuai
19	Memeriksa halaman <i>online class</i>	Menampilkan kelas sesuai tab tanggal pada hari dan tanggal yang telah dibuat oleh <i>admin</i>	Data <i>online class</i> sesuai dengan tab tanggal	Sesuai
20	Memeriksa konfirmasi Absent pada <i>Online Class</i>	Menampilkan konfirmasi terlebih dahulu sebelum melakukan presensi	Berhasil memverifikasi <i>sure/cancel</i> sebelum melakukan presensi	Sesuai
21	Memeriksa Unduh file (<i>notes</i>)	File <i>notes</i> dapat diunduh dan terbuka	File pdf terunduh dan dapat dibuka	Sesuai
22	Memeriksa Unduh latihan soal (<i>exercise</i>)	File soal dapat diunduh dan terbuka	File pdf terunduh dan dapat dibuka	Sesuai
23	Memeriksa <i>search bar</i> pada halaman (<i>explore video</i>)	Menampilkan judul video yang dicari	Setelah memasukkan judul pada <i>search bar</i> , hanya muncul judul yang sedang dicari	Sesuai
24	Memeriksa <i>view all</i> dan <i>category</i> pada halaman (<i>explore video</i>)	Menampilkan 5 <i>card category</i> dan video sesuai categorynya	Ketika memilih <i>card "Mathematics"</i> maka muncul video-video dengan <i>subject</i> matematika	Sesuai
Lanjut pada halaman berikutnya				

Tabel 3.12 Hasil Pengujian Sistem Menggunakan Metode Black-Box (lanjutan)

No.	Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
25	Putar video pembelajaran (videos)	Pengguna diarahkan ke tautan video eksternal dan video dapat diputar	Setelah klik, pengguna berhasil dialihkan ke <i>YouTube</i> dan video dapat diputar dengan lancar	Sesuai
26	Memeriksa logout	Ketika menekan tombol logout, maka muncul dialog konfirmasi <i>logout</i>	Setelah mengakses profil dan memilih logout, dialog konfirmasi <i>logout</i> muncul	Sesuai
27	Memeriksa logout dengan dialog konfirmasi	Setelah menekan logout, muncul dialog konfirmasi dengan pilihan <i>Logout</i> atau <i>Cancel</i>	Dialog konfirmasi muncul dan ketika memilih <i>Logout</i> , pengguna berhasil <i>logout</i> dan diarahkan ke halaman <i>choose role</i> dan ketika memilih <i>Cancel</i> , pengguna tetap berada di <i>dashboard</i>	Sesuai

3.6 Kendala dan Solusi yang Ditemukan

Selama proses rancang bangun sistem aplikasi di PT Matica Inovasi Edukasi, ditemukan satu kendala dari sisi *back-end*. Kendala tersebut menjadi bagian penting dalam proses pembuatan karena menuntut solusi yang adaptif agar sistem tetap berjalan dengan baik sesuai perencanaan. Penanganan terhadap kendala ini juga berkontribusi terhadap peningkatan kualitas aplikasi dan efisiensi dalam proses pengembangan.

3.6.1 Kendala

Kendala utama yang ditemukan adalah keterbatasan penggunaan *Firebase Storage*. Layanan ini, yang awalnya dirancang sebagai media penyimpanan file seperti dokumen yang dikirim oleh orang tua siswa kepada guru, ternyata hanya menyediakan kapasitas terbatas pada versi gratis. Untuk mengakses fitur unggahan file secara penuh, pengguna diwajibkan untuk melakukan peningkatan ke layanan berbayar, yang tidak memungkinkan dalam konteks pengembangan sistem ini. Kondisi ini menyebabkan fitur seperti *Send File Parent*, *Notes*, dan *Exercise* tidak dapat diimplementasikan.

Selain kendala penyimpanan, tantangan lain yang muncul adalah kebutuhan untuk mempelajari teknologi baru, yaitu integrasi sistem penyimpanan eksternal dengan Android Studio. Penggunaan layanan tambahan di luar Firebase membutuhkan pemahaman baru terkait alur autentikasi, pengelolaan file, serta cara menjembatani dua layanan berbeda agar tetap berjalan dalam satu ekosistem aplikasi yang konsisten. Hal ini menjadi hambatan tersendiri dalam proses pengembangan karena memerlukan waktu dan adaptasi terhadap dokumentasi teknis yang berbeda dari layanan sebelumnya.

3.6.2 Solusi

Sebagai solusi atas permasalahan tersebut, sistem *back-end* untuk organisir penyimpanan file dialihkan menggunakan layanan alternatif berupa Supabase. Supabase dipilih karena menyediakan penyimpanan file secara gratis dan menawarkan kemudahan integrasi dengan aplikasi berbasis Android. Melalui Supabase, file yang diunggah oleh pengguna dapat disimpan dengan aman, dan URL file tersebut kemudian dicatat dalam *Firebase Database* agar tetap terintegrasi dengan sistem yang telah dibangun. Dengan pendekatan ini, fitur unggahan file tetap dapat berjalan sesuai kebutuhan tanpa mengorbankan biaya tambahan. Penerapan solusi tersebut turut mendukung proses kerja magang agar dapat berjalan lebih efektif, terorganisir, dan sesuai dengan standar yang diharapkan oleh perusahaan.