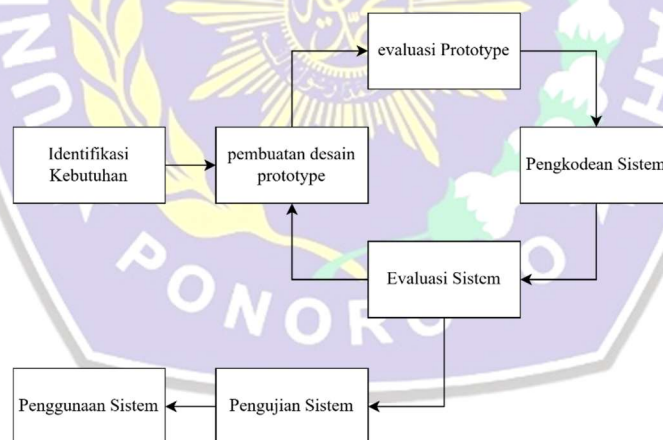


## BAB III METODE PENELITIAN

### 3.1 Tahapan Penelitian

Bab ini membahas metodologi penelitian dengan menggunakan metode *prototype*, yang terdiri dari beberapa tahapan utama, yaitu identifikasi kebutuhan, pembuatan desain *prototype*, evaluasi *prototype*, pengkodean sistem, evaluasi sistem, pengujian sistem, dan penggunaan sistem. Tahapan pertama adalah mengidentifikasi kebutuhan untuk menganalisis masalah dan menentukan spesifikasi sistem. Selanjutnya, dilakukan perancangan dan evaluasi *prototype* guna memastikan rancangan telah sesuai dengan kebutuhan. Setelah itu, sistem dikembangkan melalui tahap pengkodean, kemudian dievaluasi dan diuji untuk memastikan fungsionalitasnya berjalan optimal. Tahap akhir adalah implementasi sistem agar dapat digunakan sesuai dengan tujuan penelitian. Dalam alur penelitian yang menggunakan metode *prototype*, tahapan-tahapan tersebut diuraikan di gambar 3.1 berikut :



Gambar 3. 1 Alur Tahapan Penelitian

#### 3.2.1 Identifikasi Masalah

Identifikasi masalah pada penelitian ini adalah bahwa motivasi belajar siswa, khususnya pada materi Matematika semester genap kelas 7, sering kali rendah karena pendekatan pembelajaran yang kurang interaktif. Materi seperti perbandingan,



### 3.2.3 Alat pendukung

Alat Pendukung yang digunakan dalam penelitian ini adalah sebagai berikut:

a. Perangkat Keras Pengembangan Sistem (Hardware)

Perangkat keras yang digunakan dalam penelitian ini dengan spesifikasi sebagai berikut:

1. Laptop : *MSI*
2. Processor : *Intel(R) Core(TM) i3-1215U(8CPUs),~2.5GHz*
3. RAM : *8GB*
4. Penyimpanan: *SSD 256GB*

b. Perangkat Lunak Pengembangan Sistem (Software)

Perangkat lunak yang digunakan dalam penelitian ini dengan spesifikasi sebagai berikut:

1. Sistem Operasi : *Windows 11*
2. Bahasa Pemrograman : *Python, Javascript, ReactJs*
3. Text Editor : *Visual Studio Code*
4. Web Browser : *Google Chrome*

c. Library

*Library* atau *package* yang digunakan pada penelitian ini antara lain:

1. ReactJS

Digunakan untuk membangun antarmuka pengguna (UI) pada platform Edusmart. ReactJS mempermudah pengembangan aplikasi berbasis komponen, sehingga elemen-elemen UI seperti leaderboard, papan poin, dan tampilan badge dapat dibuat secara modular dan dinamis.

2. Express.js

Express.js digunakan sebagai framework backend yang menangani routing, logika server, dan komunikasi dengan database. Express memfasilitasi integrasi dengan frontend React dan memungkinkan pengelolaan API yang efisien.

### 3. MongoDB

MongoDB digunakan sebagai database untuk menyimpan data pengguna, seperti siswa, guru, tugas, poin, dan leaderboard. Database ini menawarkan fleksibilitas tinggi dengan format berbasis dokumen JSON, yang cocok untuk aplikasi pembelajaran interaktif.

### 4. Node.js

Node.js adalah runtime environment yang digunakan untuk menjalankan backend aplikasi Edusmart. Node.js mendukung operasi asynchronous, sehingga cocok untuk menangani banyak pengguna secara bersamaan dengan performa yang baik.

### 5. JWT (JSON Web Token)

JWT digunakan untuk mengelola autentikasi pengguna pada platform Edusmart. Library ini memastikan hanya pengguna yang sah yang dapat mengakses fitur tertentu, seperti materi pembelajaran dan leaderboard.

#### 3.2.4 Tempat penelitian

Tempat penelitian ini adalah MTs Muhammadiyah 1 Ponorogo yang berlokasi di Jl. Stadion Timur No.20 B, Ronowijayan, Kertosari, Kec. Babadan, Kabupaten Ponorogo, Jawa Timur 63491. Madrasah ini memiliki jumlah tenaga pendidik dan staf sebanyak 15 orang yang terdiri dari guru dan tenaga kependidikan. Adapun jumlah peserta didik secara keseluruhan sebanyak 72 siswa yang tersebar di kelas VII, VIII, dan IX. Kondisi tersebut menunjukkan bahwa MTs Muhammadiyah 1 Ponorogo merupakan lembaga pendidikan dengan skala menengah yang tetap mengedepankan mutu pembelajaran.

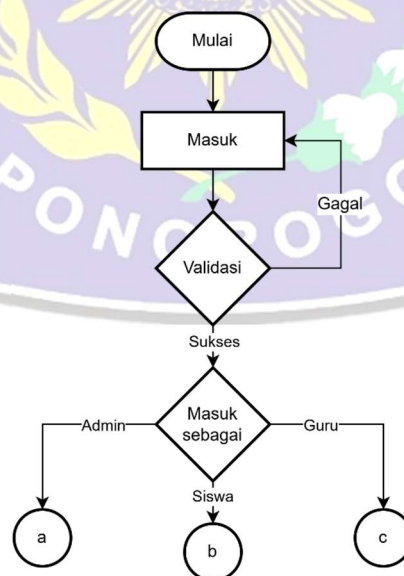
Subjek dalam penelitian ini adalah siswa kelas VII semester genap. Pemilihan kelas VII dilakukan karena pada jenjang ini siswa mulai beradaptasi dengan pola pembelajaran yang lebih kompleks dibandingkan sekolah dasar, sehingga diperlukan strategi evaluasi yang lebih sistematis. Selain itu, siswa kelas VII dinilai lebih

representatif untuk melihat efektivitas penerapan sistem evaluasi berbasis algoritma rule-based dalam meningkatkan keteraturan proses penilaian. Penelitian ini difokuskan pada implementasi platform evaluasi pembelajaran siswa dengan algoritma rule-based di MTs Muhammadiyah 1 Ponorogo. Penerapan algoritma rule-based diharapkan mampu membantu guru dalam melakukan penilaian secara otomatis, objektif, dan terstruktur sesuai dengan aturan yang telah ditetapkan. Dengan adanya sistem ini, proses evaluasi dapat berjalan lebih cepat, adil, dan memberikan umpan balik yang jelas kepada siswa mengenai capaian belajar mereka..

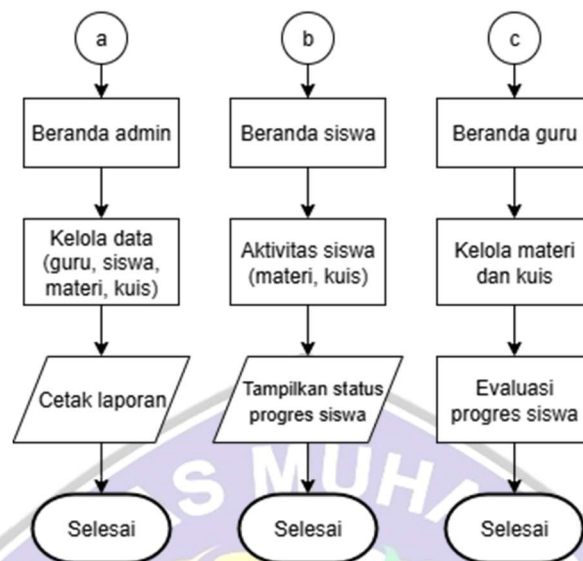
### 3.3 Metode Perancangan Prototype

#### 3.3.1 Flowchart Sistem

*Flowchart* sistem adalah representasi visual yang menggambarkan urutan langkah-langkah atau proses dalam sistem secara terstruktur. *Flowchart* ini mencakup berbagai elemen seperti keputusan, proses, dan hubungan antar proses yang membantu dalam memahami alur kerja atau operasi sistem secara jelas dan sistematis. Ditunjukkan pada gambar 3.2 dibawah.



Gambar 3. 3 *Flowchart* sistem 1



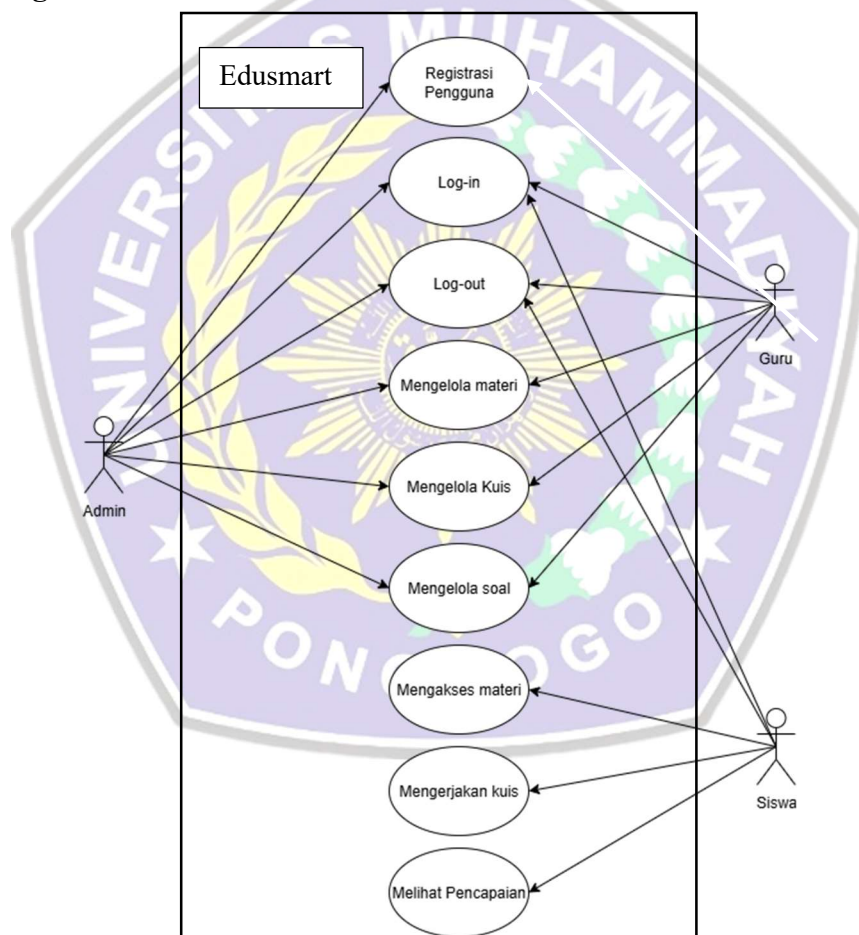
Gambar 3. 4 *Flowchart* sistem 2

Setelah berhasil melakukan autentikasi pada gambar 3.2, pengguna yang masuk sebagai admin akan diarahkan ke beranda admin. Pada gambar 3.3 admin memiliki akses untuk mengelola berbagai data penting dalam sistem, seperti data guru, data siswa, mata pelajaran, dan kelas yang tersedia. Setelah melakukan pengelolaan data, admin dapat mencetak laporan yang berisi informasi hasil pengolahan data tersebut. Proses akan dianggap selesai setelah laporan berhasil dicetak, menandakan bahwa tugas admin dalam mengelola sistem e-learning sudah terlaksana dengan baik.

Pengguna yang masuk sebagai siswa akan diarahkan ke beranda siswa setelah melalui proses autentikasi yang sukses. Di dalam sistem, siswa dapat mengakses berbagai aktivitas pembelajaran seperti mengerjakan soal dan kuis yang telah disediakan. Setelah menyelesaikan aktivitas tersebut, sistem secara otomatis akan melakukan penilaian progres belajar siswa. Hasil dari penilaian tersebut kemudian ditampilkan dalam bentuk status progres siswa yang menunjukkan pencapaian mereka dalam proses pembelajaran. Proses selesai setelah status progres siswa ditampilkan sepenuhnya.

Bagi pengguna yang masuk sebagai guru, setelah proses autentikasi berhasil, mereka akan diarahkan ke beranda guru. Di sini, guru memiliki akses untuk mengelola materi pembelajaran serta soal yang akan diberikan kepada siswa sebagai bagian dari proses pembelajaran. Setelah selesai mengelola materi dan soal, guru dapat melakukan evaluasi terhadap progres belajar siswa berdasarkan hasil pengerjaan soal atau aktivitas yang dilakukan. Proses selesai ketika evaluasi progres siswa telah dilakukan dengan lengkap.

### 3.3.2 Diagram Use Case



Gambar 3. 5 Use Case Diagram

Diagram use case sistem Edusmart melibatkan tiga aktor utama: Admin, Guru, dan Siswa, dengan peran yang berbeda yang ditunjukkan oleh gambar 3.4. Admin

bertugas mengelola Data Pengguna, Materi, dan Kuis, termasuk membuat, memperbarui, dan menghapus konten serta melakukan validasi data melalui fitur Mengelola Pengguna. Setelah selesai, Admin dapat keluar menggunakan fitur Logout.

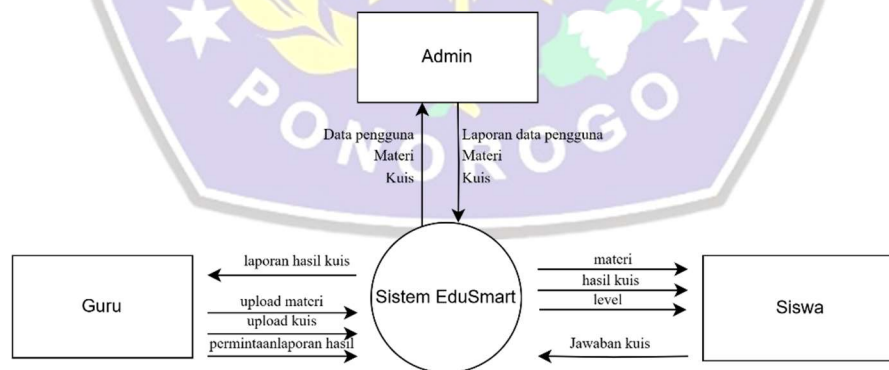
Siswa menggunakan sistem dengan fitur Login, kemudian dapat mengakses Materi dan mengerjakan Kuis yang tersedia. Setelah mengerjakan, siswa dapat melihat hasilnya melalui fitur Melihat Hasil Kuis dan memperoleh Badge jika mencapai skor tertentu. Mereka juga dapat memantau perkembangan belajar melalui fitur Melihat Pencapaian sebelum keluar dengan fitur Logout.

Guru mendukung pembelajaran dengan menggunakan fitur Login, membuat, memperbarui Materi, serta menyusun dan mengelola Kuis menggunakan fitur Mengelola Soal. Setelah menyelesaikan tugas, guru dapat keluar dengan fitur Logout.

Rancangan ini membagi peran dengan jelas, di mana Admin fokus pada pengelolaan data, Guru mengelola materi dan evaluasi, sedangkan Siswa fokus pada pembelajaran dan evaluasi hasil belajar. Sistem ini dirancang untuk mendukung proses belajar mengajar yang terorganisir dan terintegrasi.

### 3.3.3 Data Flow Diagram

#### a. DFD Level 0 (Context Diagram)



Gambar 3. 6 DFD Level 0

Diagram DFD (Data Flow Diagram) Level 0 yang ditampilkan di gambar 3.5, menggambarkan alur data dalam sistem EduSmart dengan tiga entitas utama, yaitu Admin, Guru, dan Siswa yang saling berinteraksi dengan sistem pusat.

Admin berinteraksi dengan sistem melalui beberapa alur data, seperti Login, Mengelola Data Pengguna, dan Mengelola Konten Pembelajaran. Sebagai respons, sistem mengirimkan data kembali kepada admin berupa Laporan Aktivitas Pengguna dan Data yang Telah Diperbarui, yang membantu admin dalam melakukan pengawasan dan pengelolaan sistem secara keseluruhan.

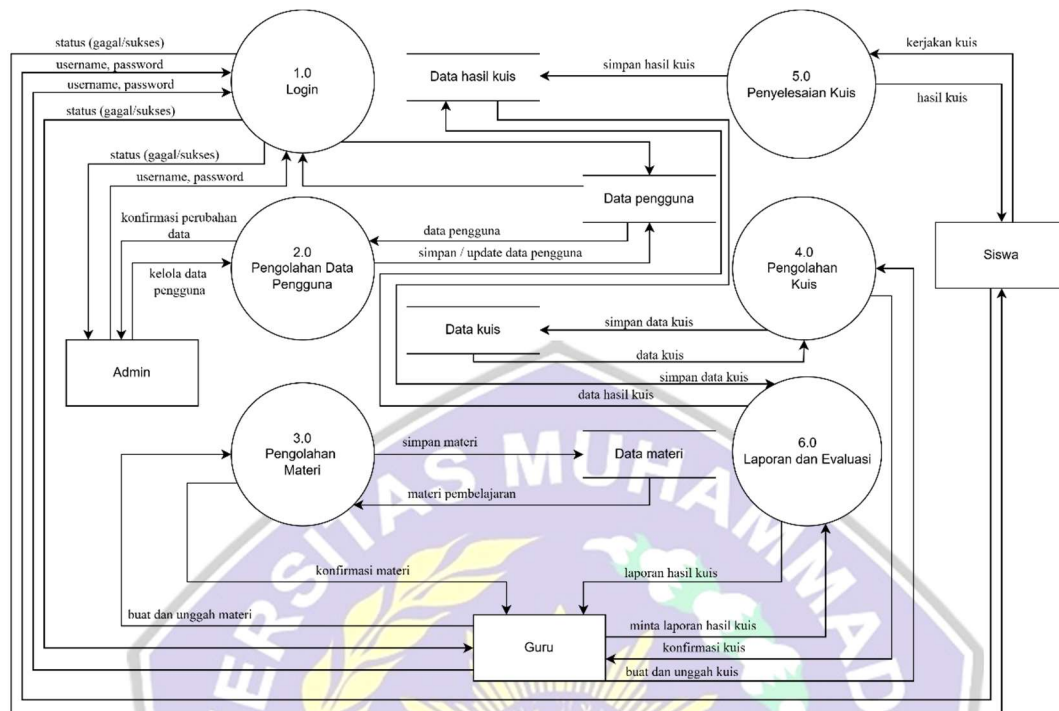
Guru menggunakan sistem untuk Login, Mengunggah Materi Pembelajaran, dan Membuat Kuis atau Tugas. Sebagai keluaran, sistem memberikan akses kepada guru untuk melihat Data Aktivitas Siswa dan Laporan Hasil Kuis, yang memungkinkan guru untuk memantau perkembangan siswa dan mengelola materi secara efektif.

Siswa berinteraksi dengan sistem melalui alur data seperti Login, Mengakses Materi Pembelajaran, Mengerjakan Kuis, dan Mengunggah Tugas. Sebagai hasil, sistem mengirimkan kembali Materi yang Diminta, Hasil Kuis, dan Umpan Balik Tugas kepada siswa, yang mendukung proses pembelajaran yang interaktif dan terarah.

Dengan alur data ini, Sistem EduSmart berfungsi sebagai pusat pengolahan data yang terintegrasi, memastikan bahwa seluruh kebutuhan Admin, Guru, dan Siswa dapat dipenuhi dalam satu platform yang efisien dan mendukung proses belajar mengajar secara optimal.



b. DFD Level 1



Gambar 3. 7 DFD Level 1

DFD Level 1 dalam sistem EduSmart ini pada gambar 3.6, menggambarkan alur data yang lebih rinci dari sistem dengan menunjukkan proses utama, entitas eksternal, dan penyimpanan data yang terlibat. Proses pertama adalah Login (1.0), di mana semua entitas, baik Admin, Guru, maupun Siswa harus melakukan login terlebih dahulu untuk mengakses sistem. Data login akan diverifikasi melalui basis data Data Pengguna untuk memastikan keabsahan akun sebelum pengguna dapat mengakses fitur yang tersedia dalam sistem.

Proses berikutnya adalah Pengolahan Data Pengguna (2.0) yang dikelola oleh Admin. Dalam proses ini, admin dapat melakukan penambahan, penghapusan, serta pembaruan data pengguna seperti nama, email, dan peran (admin, guru, siswa). Semua data pengguna yang diperbarui akan disimpan dalam basis data Data Pengguna untuk memastikan data tetap terorganisir dan akurat.

Pengolahan Materi (3.0) adalah proses yang melibatkan Guru dalam membuat, memperbarui, dan menghapus materi pembelajaran yang akan diakses oleh siswa. Materi yang dibuat guru akan disimpan dalam Data Materi dan dapat diakses oleh siswa untuk mendukung proses belajar. Materi yang dikelola dapat berupa teks, gambar, atau file pendukung lainnya.

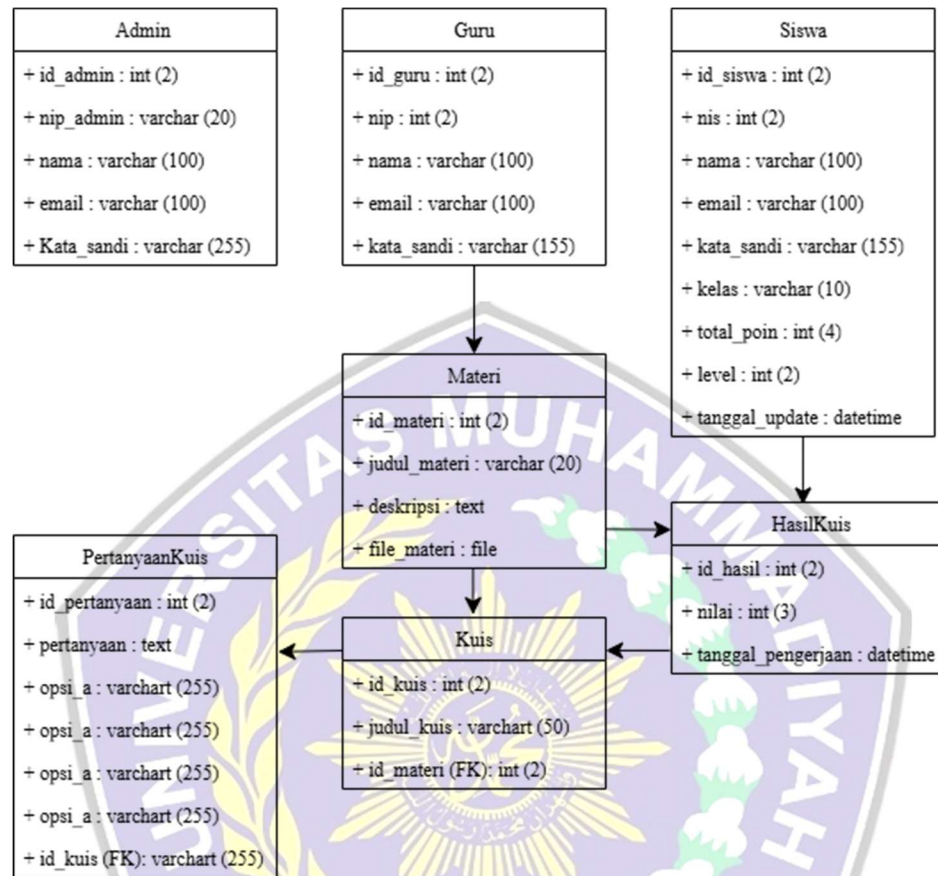
Pengolahan Kuis (4.0) merupakan proses di mana Guru membuat kuis dengan mengunggah pertanyaan, pilihan jawaban, serta menentukan kunci jawaban. Data yang diinput guru akan disimpan dalam Data Kuis dan dapat diakses oleh siswa untuk dikerjakan. Proses ini memungkinkan guru untuk menilai pemahaman siswa melalui aktivitas evaluasi berbasis kuis.

Penyelesaian Kuis (5.0) melibatkan Siswa yang mengerjakan kuis yang telah disediakan dalam sistem. Setelah siswa menyelesaikan kuis, data hasil pengerjaan seperti nilai dan tanggal pengerjaan akan disimpan dalam Data Hasil Kuis sebagai dokumentasi pencapaian siswa.

Proses terakhir adalah Laporan dan Evaluasi (6.0) yang memungkinkan Admin dan Guru mengakses laporan hasil belajar siswa yang tersimpan dalam Data Hasil Kuis. Laporan ini digunakan untuk mengevaluasi performa siswa berdasarkan hasil kuis dan materi yang telah dipelajari.

DFD Level 1 ini memperjelas bagaimana setiap entitas berinteraksi dengan sistem EduSmart secara terstruktur, di mana Admin fokus pada pengelolaan data pengguna, Guru pada pembuatan materi dan evaluasi, serta Siswa dalam proses pembelajaran dan evaluasi hasil belajar.

### 3.3.4 Entity Relationship Diagram



Gambar 3. 8 Perancangan UML

Diagram UML di atas menggambarkan struktur sistem pembelajaran digital yang mengelola data pengguna (admin, guru, siswa), materi pembelajaran, kuis, serta hasil pengerjaan kuis. Sistem ini memungkinkan guru untuk menyusun materi dan kuis, yang kemudian diakses oleh siswa sebagai bentuk interaksi pembelajaran. Setiap siswa memiliki data poin, level, dan histori hasil kuis sebagai bagian dari evaluasi algoritma rule-based, sementara admin berperan dalam pengelolaan keseluruhan sistem.

| Entitas | Atribut Penting                              | Catatan                   |
|---------|----------------------------------------------|---------------------------|
| Admin   | id_admin, nip_admin, nama, email, kata_sandi | Bertugas mengelola sistem |

|                |                                                                           |                                                           |
|----------------|---------------------------------------------------------------------------|-----------------------------------------------------------|
| Guru           | id_guru, nip, nama, email, kata_sandi                                     | Pengunggah materi dan pembuat kuis                        |
| Siswa          | id_siswa, nis, nama, email, kata_sandi, total_poin, level, tanggal_update | Pengguna utama yang mengakses materi dan mengerjakan kuis |
| Materi         | id_materi, judul_materi, deskripsi, file_materi                           | Konten pembelajaran digital                               |
| Kuis           | id_kuis, judul_kuis, id_materi (FK)                                       | Set soal yang terhubung ke materi tertentu                |
| PertanyaanKuis | id_pertanyaan, pertanyaan, opsi_a-d, id_kuis (FK)                         | Isi dari kuis                                             |
| HasilKuis      | id_hasil, nilai, tanggal_pengerjaan                                       | Catatan nilai siswa                                       |

#### Relasi Antar Entitas

a. Materi ↔ Kuis

Setiap kuis terhubung ke satu materi melalui id\_materi. Artinya, kuis hanya berlaku untuk materi tertentu.

b. Kuis ↔ PertanyaanKuis

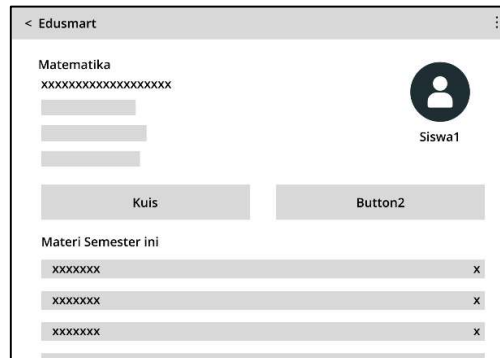
Kuis terdiri dari banyak pertanyaan. Relasi ini menghubungkan daftar soal ke kuis melalui foreign key id\_kuis.

c. Siswa ↔ HasilKuis

Mencatat interaksi siswa dengan kuis. Sistem menyimpan nilai serta waktu pengerjaan.

### 3.3.5 User Interface

Perancangan *user interface* pada sistem ini dirancang untuk memberikan pengalaman pengguna yang nyaman dan mudah dipahami dalam aktifitas pembelajaran siswa.

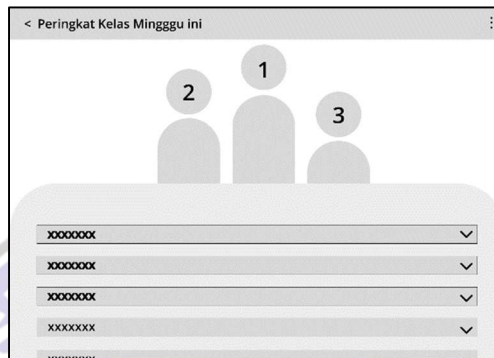


Pada gambar 3.8 halaman ini menggambarkan desain halaman mata pelajaran dalam sebuah platform pembelajaran, terdapat judul mata pelajaran yang sedang diakses. Dibawahnya terdapat bagian yang menampilkan nilai kuis yang telah diperoleh oleh pengguna dalam mata pelajaran tersebut, yang diikuti dengan tombol interaktif berupa "Button Kuis" untuk mengakses atau memulai kuis dan "Button Latihan" untuk mengerjakan latihan soal. Di bagian bawah halaman, terdapat daftar bab dalam bentuk dropdown atau expandable list yang dapat diklik untuk menampilkan materi pelajaran secara terstruktur. Desain ini dirancang untuk memberikan kemudahan dalam mengakses informasi mata pelajaran, memantau capaian belajar, dan mengakses aktivitas seperti kuis serta latihan soal dengan terorganisir



Wireframe yang ditampilkan gambar 3.9 merupakan desain halaman pengerjaan kuis interaktif. Di tengah halaman, terdapat area yang menampilkan soal kuis dengan

format pilihan ganda, di mana beberapa pilihan jawaban tersusun rapi dalam bentuk kotak yang dapat diklik. Di bagian bawah, terdapat dua tombol navigasi, yaitu Preview untuk kembali ke soal sebelumnya dan Next untuk melanjutkan ke soal berikutnya. Desain ini dirancang untuk memberikan pengalaman yang sederhana, terorganisir, dan mudah digunakan bagi peserta dalam mengerjakan kuis secara interaktif.



Gambar 3. 11 Tampilan leaderboard

*Wireframe* yang ditampilkan gambar 3.10 merupakan desain halaman *leaderboard* dalam sebuah platform pembelajaran yang menampilkan peringkat siswa berdasarkan pencapaian tertinggi. Bagian utama halaman menampilkan podium dengan tiga posisi teratas, di mana peringkat pertama berada di tengah dengan posisi tertinggi, diapit oleh peringkat kedua di sisi kiri dan peringkat ketiga di sisi kanan dengan ketinggian podium yang lebih rendah. Ini memberikan penekanan visual pada peringkat tertinggi.

Di bawah podium, terdapat daftar peringkat siswa yang menampilkan lebih banyak peserta di luar tiga besar. Setiap baris dalam daftar mencakup nomor peringkat, foto profil atau avatar siswa, jumlah bintang yang mencerminkan pencapaian atau skor, serta ikon panah yang menunjukkan peringkat siswa naik atau turun.

Desain ini dirancang untuk memberikan tampilan yang jelas dan menarik tentang pencapaian siswa, memotivasi mereka untuk bersaing secara positif, serta memberikan penghargaan visual kepada siswa dengan performa terbaik dalam platform pembelajaran.

### 3.3.6 Perancangan Database

Perancangan database bertujuan untuk mendukung implementasi sistem Edusmart, dengan memastikan penyimpanan data dilakukan secara terstruktur, aman, dan dapat diakses dengan efisien. Berikut adalah tabel-tabel yang dirancang untuk sistem Edusmart beserta deskripsinya.

#### a. Tabel pengguna

Tabel ini digunakan untuk menyimpan data pengguna sistem, baik itu Administrator, Guru, maupun Siswa.

Tabel 3. 1 Struktur tabel pengguna

| Kolom            | Tipe Data                      | Keterangan            |
|------------------|--------------------------------|-----------------------|
| id_pengguna (PK) | INT<br>AUTO_INCREMENT          | Primary Key           |
| nama             | VARCHAR(100)                   | Nama pengguna         |
| email            | VARCHAR(100)                   | Email pengguna (unik) |
| kata_sandi       | VARCHAR(255)                   | Kata sandi (hash)     |
| role             | ENUM('Admin', 'Guru', 'Siswa') | Peran pengguna        |
| tanggal_dibuat   | DATETIME                       | Tanggal akun dibuat   |

#### b. Tabel admin

Menyimpan data pengguna dengan peran admin, terdiri dari kolom *id\_admin* (kunci utama), *id\_pengguna* (kunci asing ke tabel Pengguna), dan *nip\_admin*.

Tabel 3. 2 Struktur tabel admin

| Kolom            | Tipe Data             | Keterangan                    |
|------------------|-----------------------|-------------------------------|
| id_admin (PK)    | INT<br>AUTO_INCREMENT | Primary Key                   |
| id_pengguna (FK) | INT                   | Foreign Key ke tabel Pengguna |

|           |             |                           |
|-----------|-------------|---------------------------|
| nip_admin | VARCHAR(20) | Nomor Induk Pegawai Admin |
|-----------|-------------|---------------------------|

c. Tabel guru

berisi data pengguna berperan sebagai guru, mencakup *id\_guru* (kunci utama), *id\_pengguna* (kunci asing), *nip*, dan *mapel*. Terhubung ke tabel Pengguna.

Tabel 3. 3 Struktur tabel guru

| Kolom            | Tipe Data             | Keterangan                    |
|------------------|-----------------------|-------------------------------|
| id_guru (PK)     | INT<br>AUTO_INCREMENT | Primary Key                   |
| id_pengguna (FK) | INT                   | Foreign Key ke tabel Pengguna |
| nip              | VARCHAR(20)           | Nomor Induk Pegawai Guru      |
| mapel            | VARCHAR(50)           | Mata pelajaran yang diajarkan |

d. Tabel siswa

Menyimpan data pengguna berperan siswa, dengan kolom *id\_siswa* (kunci utama), *id\_pengguna* (kunci asing), *nis*, dan *kelas*.

Tabel 3. 4 Struktur tabel siswa

| Kolom            | Tipe Data             | Keterangan                     |
|------------------|-----------------------|--------------------------------|
| id_siswa (PK)    | INT<br>AUTO_INCREMENT | Primary Key                    |
| id_pengguna (FK) | INT                   | Foreign Key ke tabel Pengguna  |
| nis              | VARCHAR(20)           | Nomor Induk Siswa              |
| kelas            | VARCHAR(10)           | Kelas siswa (misal: "X IPA 1") |

e. Tabel materi

Berisi data materi pembelajaran dengan kolom *id\_materi* (kunci utama), *judul\_materi*, *deskripsi*, *file\_materi*, dan *id\_guru* (kunci asing ke tabel Guru).

Tabel 3. 5 Struktur tabel materi

| Kolom                 | Tipe Data          | Keterangan                |
|-----------------------|--------------------|---------------------------|
| <i>id_materi</i> (PK) | INT AUTO_INCREMENT | Primary Key               |
| <i>judul_materi</i>   | VARCHAR(255)       | Judul materi              |
| <i>deskripsi</i>      | TEXT               | Deskripsi materi          |
| <i>file_materi</i>    | VARCHAR(255)       | Path atau URL file materi |
| <i>id_guru</i> (FK)   | INT                | Foreign Key ke tabel Guru |

f. Tabel kuis

Menyimpan data kuis, meliputi *id\_kuis* (kunci utama), *judul\_kuis*, *id\_materi* (kunci asing ke tabel Materi), dan *id\_guru* (kunci asing ke tabel Guru).

Tabel 3. 6 Struktur tabel kuis

| Kolom                 | Tipe Data          | Keterangan                  |
|-----------------------|--------------------|-----------------------------|
| <i>id_kuis</i> (PK)   | INT AUTO_INCREMENT | Primary Key                 |
| <i>judul_kuis</i>     | VARCHAR(255)       | Judul kuis                  |
| <i>id_materi</i> (FK) | INT                | Foreign Key ke tabel Materi |
| <i>id_guru</i> (FK)   | INT                | Foreign Key ke tabel Guru   |

g. Tabel pertanyaan\_kuis

Berisi data pertanyaan kuis, mencakup *id\_pertanyaan* (kunci utama), *id\_kuis* (kunci asing), *pertanyaan*, *opsi\_a-d*, dan *jawaban\_benar*. Terhubung ke tabel Kuis.

Tabel 3. 7 Struktur tabel pertanyaan\_kuis

| Kolom                     | Tipe Data          | Keterangan  |
|---------------------------|--------------------|-------------|
| <i>id_pertanyaan</i> (PK) | INT AUTO_INCREMENT | Primary Key |

|               |                          |                           |
|---------------|--------------------------|---------------------------|
| id_kuis (FK)  | INT                      | Foreign Key ke tabel Kuis |
| pertanyaan    | TEXT                     | Teks pertanyaan           |
| opsi_a        | VARCHAR(255)             | Pilihan jawaban A         |
| opsi_b        | VARCHAR(255)             | Pilihan jawaban B         |
| opsi_c        | VARCHAR(255)             | Pilihan jawaban C         |
| opsi_d        | VARCHAR(255)             | Pilihan jawaban D         |
| jawaban_benar | ENUM('A', 'B', 'C', 'D') | Jawaban yang benar        |

h. Tabel hasil\_kuis

Menyimpan hasil pengerjaan kuis siswa, dengan *id\_hasil* (kunci utama), *id\_kuis* (kunci asing), *id\_siswa* (kunci asing), *nilai*, dan *tanggal\_pengerjaan*.

Tabel 3. 8 Struktur tabel hasil\_kuis

| Kolom              | Tipe Data             | Keterangan                 |
|--------------------|-----------------------|----------------------------|
| id_hasil (PK)      | INT<br>AUTO_INCREMENT | Primary Key                |
| id_kuis (FK)       | INT                   | Foreign Key ke tabel Kuis  |
| id_siswa (FK)      | INT                   | Foreign Key ke tabel Siswa |
| nilai              | FLOAT                 | Nilai kuis                 |
| tanggal_pengerjaan | DATETIME              | Tanggal pengerjaan kuis    |

i. Tabel poin\_level

Menyimpan data poin dan level siswa, meliputi *id\_poin* (kunci utama), *id\_siswa* (kunci asing), *total\_poin*, *level*, dan *tanggal\_update*.

Tabel 3. 9 Struktur tabel poin\_level

| Kolom         | Tipe Data          | Keterangan                 |
|---------------|--------------------|----------------------------|
| id_poin (PK)  | INT AUTO_INCREMENT | Primary Key                |
| id_siswa (FK) | INT                | Foreign Key ke tabel Siswa |

|                |             |                              |
|----------------|-------------|------------------------------|
| total_poin     | INT         | Total poin yang diperoleh    |
| level          | VARCHAR(50) | Level siswa berdasarkan poin |
| tanggal_update | DATETIME    | Tanggal pembaruan poin       |

j. Relasi antar Tabel

1. Relasi antara Tabel Pengguna dan Tabel Hasil Kuis

Kolom id\_pengguna pada tabel Hasil Kuis menjadi foreign key yang mengacu pada primary key id\_pengguna di tabel Pengguna. Relasi ini memastikan setiap hasil kuis dikaitkan dengan pengguna tertentu, baik itu siswa, guru, maupun admin.

2. Relasi antara Tabel Siswa dan Tabel Hasil Kuis

Kolom id\_siswa pada tabel Hasil Kuis menjadi foreign key yang mengacu pada primary key id\_siswa di tabel Siswa. Relasi ini memastikan bahwa data hasil kuis hanya dikaitkan dengan siswa yang mengikuti kuis tersebut.

3. Relasi antara Tabel Materi dan Tabel Kuis

Kolom id\_materi pada tabel Kuis menjadi foreign key yang mengacu pada primary key id\_materi di tabel Materi. Relasi ini memastikan bahwa setiap kuis dikaitkan dengan materi tertentu yang dibuat oleh guru.

4. Relasi antara Tabel Guru dan Tabel Materi

Kolom id\_guru pada tabel Materi menjadi foreign key yang mengacu pada primary key id\_guru di tabel Guru. Relasi ini memastikan bahwa setiap materi pembelajaran dikaitkan dengan guru yang membuatnya.

5. Relasi antara Tabel Guru dan Tabel Kuis

Kolom id\_guru pada tabel Kuis menjadi foreign key yang mengacu pada primary key id\_guru di tabel Guru. Relasi ini memastikan bahwa kuis yang dibuat selalu dikaitkan dengan guru yang bertanggung jawab.

6. Relasi antara Tabel Kuis dan Tabel Pertanyaan Kuis

Kolom id\_kuis pada tabel Pertanyaan Kuis menjadi foreign key yang mengacu pada primary key id\_kuis di tabel Kuis. Relasi ini memastikan bahwa setiap pertanyaan dikaitkan dengan kuis tertentu.

7. Relasi antara Tabel Siswa dan Tabel Poin dan Level

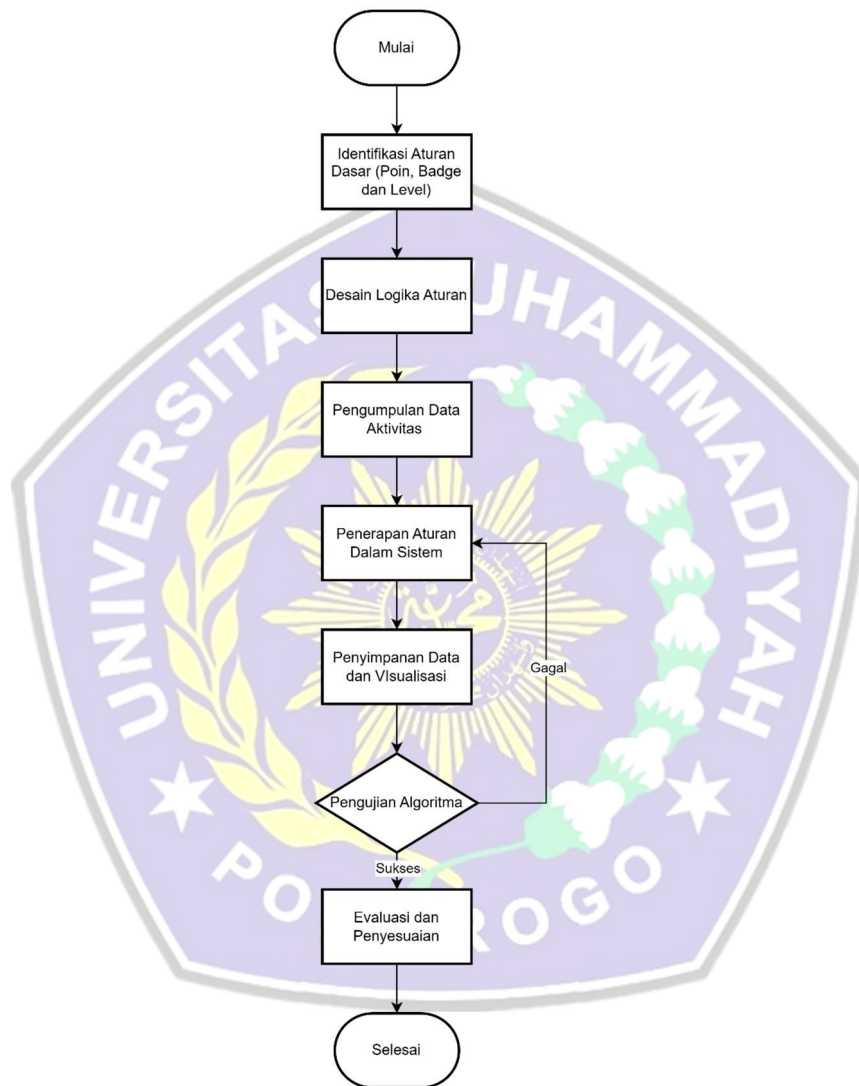
Kolom id\_siswa pada tabel Poin dan Level menjadi foreign key yang mengacu pada primary key id\_siswa di tabel Siswa. Relasi ini memastikan bahwa setiap poin dan level dikaitkan dengan siswa yang berpartisipasi dalam sistem pembelajaran CBT.

Database ini dirancang untuk memastikan integritas data melalui relasi antar tabel menggunakan Primary Key dan Foreign Key, sehingga mendukung proses pembelajaran CBT yang interaktif dan terstruktur di Edusmart.



### 3.3.7 Algoritma *Rule-Based*

Pada penelitian ini, algoritma *rule-based* digunakan untuk menentukan pemberian poin, *badge*, dan level kepada siswa berdasarkan interaksi mereka dengan platform Edusmart.



Gambar 3. 12 Flowchart algoritma

#### a. Identifikasi Aturan Dasar

Aturan ini dirancang agar penghargaan yang diberikan mencerminkan kinerja dan usaha siswa secara adil. Poin diberikan kepada siswa berdasarkan penyelesaian aktivitas seperti menjawab kuis, menyelesaikan tugas, atau

berpartisipasi dalam kompetisi, di mana jumlah poin yang dijadikan patokan adalah rata-rata yang dihitung menggunakan rumus (2.2)

$$P = \frac{\sum_{i=1}^n P_i}{n}$$

Sebagai contoh implementasi sistem penghargaan, seorang siswa yang menyelesaikan kuis dengan nilai tinggi, misalnya 90, akan memperoleh 90 poin dan mendapatkan tambahan bonus apabila kuis tersebut diselesaikan tepat waktu. Sistem pemberian badge dirancang untuk memberikan motivasi tambahan kepada siswa melalui bentuk penghargaan visual yang menarik. Badge diberikan berdasarkan total poin akumulatif yang diperoleh oleh siswa selama proses pembelajaran. Misalnya, seorang siswa yang telah mengumpulkan 120 poin akan mendapatkan badge perak, sementara siswa dengan perolehan 220 poin akan menerima badge emas. Selain itu, badge juga dapat diberikan atas pencapaian tertentu, seperti berhasil menyelesaikan seluruh kuis dalam satu materi atau meraih skor sempurna dalam kuis tertentu. Pendekatan ini bertujuan untuk mendorong siswa agar lebih aktif, konsisten, dan termotivasi dalam mengikuti seluruh tahapan pembelajaran di platform.

Sistem leveling digunakan untuk memberikan insentif tambahan kepada siswa agar terus belajar. Kenaikan level ditentukan oleh total poin yang diperoleh siswa menggunakan rumus (2.6):

$$L = \left\lceil \frac{P}{100} \right\rceil + 1$$

Di mana  $L$  adalah level siswa, dan  $P$  adalah total poin yang diperoleh. Sebagai contoh, seorang siswa dengan 350 poin akan berada pada level 4, memungkinkan dia untuk mengakses materi baru atau mendapatkan bonus tambahan.

b. Desain Logika Aturan

Logika aturan pada sistem Edusmart diimplementasikan menggunakan percabangan kondisi (*if-else*) yang dikombinasikan dengan operator logika untuk mengevaluasi aktivitas siswa. Sistem secara otomatis memeriksa data seperti nilai kuis (*score ratio*), status penyelesaian materi, dan total poin yang diperoleh. Operator logika yang digunakan antara lain:

1. **&& (AND)** – Digunakan untuk memastikan bahwa dua atau lebih kondisi harus terpenuhi secara bersamaan. Contoh: memastikan *totalPoints* tidak bernilai null **dan** nilainya minimal 0.
2. **|| (OR)** – Digunakan untuk memastikan bahwa minimal satu dari beberapa kondisi bernilai benar. Contoh: melanjutkan ke materi berikut jika nilai kuis minimal 75 **atau** materi telah diselesaikan.
3. **! (NOT)** – Digunakan untuk membalik nilai logika suatu kondisi. Contoh: menampilkan saran pembelajaran jika nilai kuis **tidak** mencapai batas minimum.

Dengan kombinasi *if-else* dan operator logika ini, sistem dapat menjalankan aturan berbasis kondisi (rule-based) untuk menentukan tindakan seperti memberikan rekomendasi materi, menaikkan level siswa, atau memberikan badge sesuai pencapaian.

c. Pengumpulan data Aktivitas siswa

Sistem memonitor dan mencatat data aktivitas siswa secara real-time, seperti nilai kuis, jumlah tugas yang diselesaikan, dan partisipasi dalam kegiatan platform.

d. Penerapan Aturan Dalam Sistem

Aturan yang telah ditetapkan diterapkan langsung ke sistem menggunakan logika pemrograman. Contohnya, jika siswa mencapai poin tertentu, sistem secara otomatis memberikan badge atau menaikkan level.

e. Penyimpanan data dan visualisasi

Hasil penghargaan berupa poin, badge, dan level disimpan dalam basis data dan ditampilkan pada profil siswa untuk memberikan umpan balik visual yang dapat memotivasi mereka.

f. Pengujian Algoritma

Pengujian algoritma rule-based dalam penelitian ini dilakukan menggunakan unit testing untuk memastikan setiap aturan berjalan sesuai dengan spesifikasi yang telah ditentukan. Unit testing menguji setiap fungsi algoritma secara terpisah, seperti pemberian poin, badge, dan kenaikan level berdasarkan interaksi siswa.

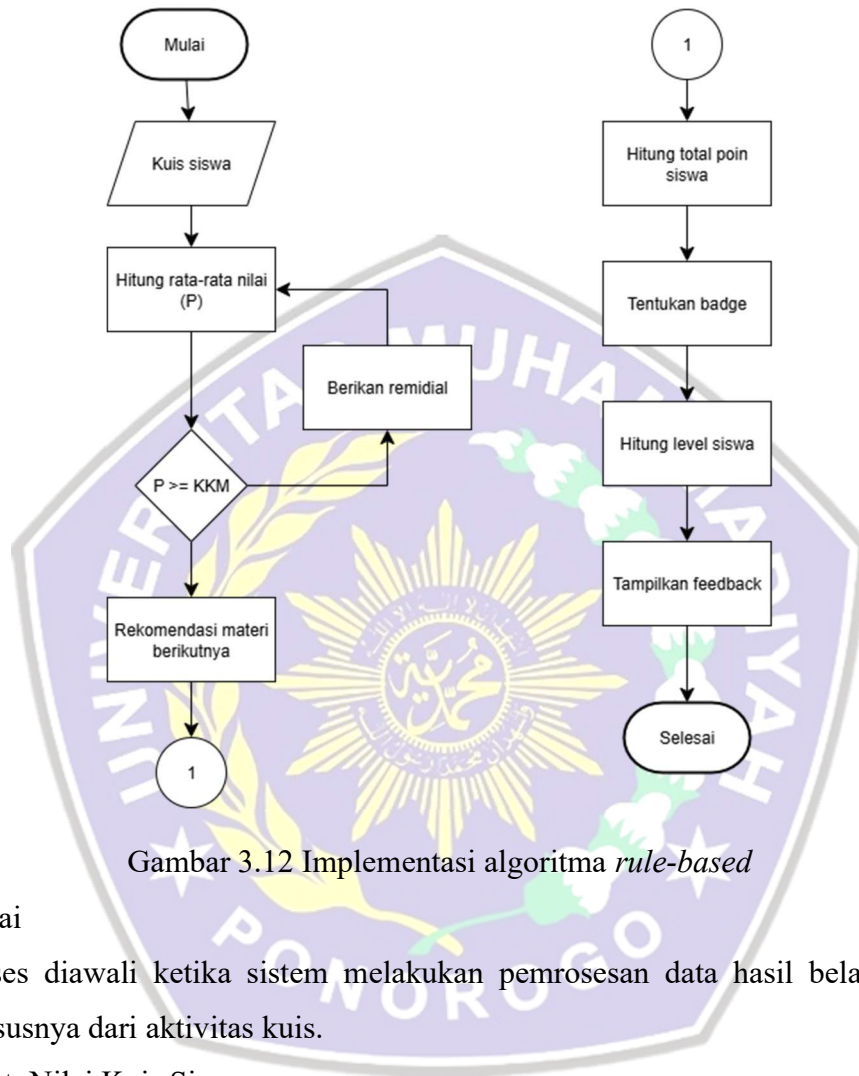
g. Evaluasi dan Penyesuaian

Tahap akhir adalah mengevaluasi hasil penerapan algoritma untuk melihat apakah penghargaan yang diberikan sudah mencerminkan usaha siswa dengan tepat. Jika diperlukan, aturan dapat disesuaikan untuk meningkatkan keadilan dan efektivitas.

Dengan penerapan ini, platform Edusmart diharapkan mampu meningkatkan motivasi dan keterlibatan siswa melalui penghargaan yang terstruktur dan terukur.

### 3.3.8 Implementasi Algoritma Rule-Based

Berikut alur implementasi algoritma rule-based pada sistem



Gambar 3.12 Implementasi algoritma *rule-based*

a. Mulai

Proses diawali ketika sistem melakukan pemrosesan data hasil belajar siswa, khususnya dari aktivitas kuis.

b. Input: Nilai Kuis Siswa

Sistem mengambil data nilai kuis siswa dari basis data. Nilai ini bisa berasal dari beberapa kuis yang telah dikerjakan.

c. Proses: Hitung Rata-rata Nilai (P)

Sistem menghitung nilai rata-rata dari semua kuis yang telah dikerjakan siswa

menggunakan rumus:  $P = \frac{\sum \text{nilai kuis}}{\text{jumlah kuis}}$

d. Percabangan: Apakah  $P \geq KKM$ ?

Sistem membandingkan nilai rata-rata dengan KKM kelulusan misalnya 70 atau 75.

1. Jika Ya (Lulus). Siswa dianggap memahami materi dan sistem akan memberikan rekomendasi materi berikutnya.
2. Jika Tidak (Tidak Lulus): Sistem memberikan remedial, yaitu berupa pengulangan materi atau latihan tambahan.

e. Hitung Total Poin Siswa

Sistem menjumlahkan seluruh poin yang diperoleh siswa dari berbagai aktivitas (kuis, tugas, presensi, dll).

Misalnya: Total Poin = Poin kuis + Poin tugas + Poin lainnya

f. Tentukan Badge

Berdasarkan total poin atau pencapaian tertentu, sistem memberikan badge atau penghargaan simbolik.

g. Hitung Level Siswa

Level menunjukkan sejauh mana progres siswa dalam sistem. Level ini bisa dihitung berdasarkan jumlah materi yang telah diselesaikan atau total poin.

h. Tampilkan Feedback

Sistem menyajikan umpan balik (feedback) yang bersifat deskriptif. Contohnya:

“Selamat! Anda telah mencapai level 2. Materi selanjutnya telah terbuka.”

“Nilai Anda masih di bawah ambang batas. Silakan ulangi materi bagian 2.”

i. Selesai

Proses evaluasi berakhir setelah sistem menampilkan feedback dan hasil analisis performa siswa.

### 3.4 Pengujian sistem

Penelitian ini menggunakan metode *Black Box Testing* untuk menguji fungsionalitas sistem tanpa memperhatikan struktur internalnya. Pengujian bertujuan untuk memastikan algoritma *Rule-Based* berjalan sesuai dengan aturan yang telah ditetapkan, menguji sistem menghasilkan output sesuai berdasarkan input yang diberikan. Skenario pengujian mencakup fitur utama sistem, seperti pemberian poin pada kuis, validasi jawaban, penyesuaian level atau penghargaan dan penentuan peringkat (*leaderboard*). Setiap skenario pengujian mencakup proses input, 50 pengolahan, dan output yang diharapkan, untuk mengevaluasi performa dan kendalan sistem yang dikembangkan.

Salah satu aspek penting dalam pengujian adalah mengukur akurasi sistem. Akurasi sistem dapat dihitung dengan menggunakan rumus (2.7) berikut:

$$\text{Akurasi sistem} = \frac{\text{Hasil benar}}{\text{Total Pengujian}} \times 100\%$$

Dengan menggunakan rumus ini, kita dapat mengevaluasi sejauh mana sistem mampu memberikan umpan balik yang tepat dan akurat pada setiap skenario pengujian. Berikut adalah tabel skenario pengujian untuk fitur utama :

Tabel 3. 10 Skenario Pengujian

| No | Fitur yang Diuji | Skenario Pengujian                                | Input                   | Output yang Diharapkan                      |
|----|------------------|---------------------------------------------------|-------------------------|---------------------------------------------|
| 1  | Login            | Pengguna memasukkan email dan password yang benar | Email<br>password valid | & Berhasil login dan diarahkan ke dashboard |

| No | Fitur yang Diuji | Skenario Pengujian                                                   | Input                      | Output yang Diharapkan                                                  |
|----|------------------|----------------------------------------------------------------------|----------------------------|-------------------------------------------------------------------------|
| 2  | Login            | Pengguna memasukkan email atau password salah                        | Email/password tidak valid | Pesan error: "Login gagal"                                              |
| 3  | Logout           | Pengguna menekan tombol logout                                       | Klik tombol logout         | Sistem menghapus sesi dan kembali ke halaman login                      |
| 4  | Dashboard Siswa  | Sistem menampilkan data statistik siswa yang baru pertama login      | Token pengguna valid       | Tampilan Level = 1, Total Poin = 0, Badge, dsb                          |
| 5  | Dashboard Siswa  | Sistem menampilkan data statistik siswa yang telah mengerjakan kuis. | Jawaban benar              | Tampilan level = 2, Performa = rendah, Total poin = 100, Badge = bronze |
| 6  | Kuis             | Siswa menjawab kuis dengan jawaban benar                             | Jawaban benar              | Poin bertambah sesuai bobot (misal 10 poin)                             |
| 7  | Tambah Materi    | Guru meninput materi tidak lengkap                                   | Kosong                     | Muncul peringatan untuk mengisi kelengkapan materi                      |
| 8  | Tambah Materi    | Guru menginput materi lengkap                                        | Judul + Materi             | Materi tersimpan dan muncul di daftar.                                  |
| 9  | Badge & Level    | Total poin siswa mencapai 100                                        | Total poin: 85             | Sistem menaikkan level, Badge = silver                                  |
| 10 | Leaderboard      | Beberapa siswa memiliki skor berbeda                                 | Skor: 50, 30, 40           | Leaderboard diurutkan: 50 → 40 → 30                                     |

Tabel 3.5 menunjukkan skenario pengujian untuk fitur utama sistem EduSmart. Pengujian dilakukan secara sistematis untuk memastikan setiap fitur berjalan sesuai spesifikasi, mulai dari tahap input data hingga hasil akhir yang diharapkan.

### 3.5 Pengujian Algoritma *Rule-Based*

Pengujian algoritma Rule-Based pada sistem EduSmart bertujuan untuk memastikan bahwa sistem dapat menghasilkan keputusan otomatis yang sesuai dengan aturan-aturan yang telah ditetapkan. Pengujian dilakukan menggunakan metode White Box Testing, yaitu metode pengujian yang difokuskan pada fungsionalitas sistem berdasarkan input dan output, tanpa memperhatikan struktur internal program. Beberapa aspek yang akan diuji dalam penerapan algoritma Rule-Based meliputi:

- a. Klasifikasi Performa Siswa, Sistem akan diuji untuk mengetahui apakah dapat mengkategorikan performa siswa berdasarkan nilai rata-rata dari seluruh kuis yang telah dikerjakan. Kategori yang digunakan antara lain: *Tinggi, Sedang, Rendah, dan Belum Ada*.
- b. Penentuan Kenaikan Level, Pengujian dilakukan untuk memastikan sistem dapat menghitung level siswa berdasarkan total poin yang diperoleh. Setiap kenaikan level ditentukan berdasarkan ambang batas tertentu (misalnya setiap 75 poin).
- c. Pemberian Badge, Sistem akan diuji untuk memberikan badge atau penghargaan simbolik kepada siswa berdasarkan pencapaian tertentu, seperti jumlah poin, penyelesaian materi, atau keaktifan.
- d. Rekomendasi Materi Berikutnya, Setelah siswa menyelesaikan kuis dan memperoleh nilai di atas ambang batas Kriteria Ketuntasan Minimal (KKM), sistem akan memberikan rekomendasi untuk melanjutkan ke materi berikutnya.
- e. Rekomendasi Remedial, Jika siswa memperoleh nilai di bawah KKM, sistem akan memberikan rekomendasi berupa pengulangan materi atau latihan tambahan untuk membantu pemahaman siswa.
- f. Pemberian Badge Loyalitas, Pengujian dilakukan untuk memastikan sistem dapat mendeteksi keaktifan siswa dalam platform, seperti sering login, menyelesaikan

materi secara konsisten, dan aktif mengerjakan kuis, kemudian memberikan badge loyalitas sesuai ketentuan.

Seluruh pengujian ini bertujuan untuk mengevaluasi apakah algoritma Rule-Based mampu menjalankan fungsinya dengan benar dalam konteks pembelajaran adaptif berbasis kuis pada platform EduSmart.

