

**RANCANG BANGUN APLIKASI PEMBELAJARAN INFORMATIKA  
BERBASIS WEB MENGGUNAKAN ALGORITMA *QUICK SORT*  
(STUDI KASUS SMP PLUS MULTAZAM)**

**SKRIPSI**

**Karya Tulis sebagai syarat memperoleh  
Gelar Sarjana Komputer dari Fakultas Teknologi Informasi  
Universitas Bale Bandung**

Disusun oleh:

**HASBI KURNIA KARIM  
NPM. 3012100045**



**PROGRAM STRATA 1  
PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG  
BANDUNG  
2025**

**LEMBAR PERSETUJUAN PEMBINGBING**

**RANCANG BANGUN APLIKASI PEMBELAJARAN INFORMATIKA  
BERBASIS WEB MENGGUNAKAN ALGORITMA *QUICK SORT*  
(STUDI KASUS SMP PLUS MULTAZAM)**

Disusun oleh:

HASBI KURNIA KARIM

NPM. 301210045

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

**Pada**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TENOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2025

Disetujui oleh:

Pembimbing utama

Pembimbing pedamping

Rosmalina, S.T., M.Kom.

NIK. 04104808122

Cecep Suwandana, S.Si., M.Kom.

NIK. 0423088802

**LEMBAR PERSETUJUAN PENGUJI**

**RANCANG BANGUN APLIKASI PEMBELAJARAN INFORMATIKA  
BERBASIS WEB MENGGUNAKAN ALGORITMA *QUICK SORT*  
(STUDI KASUS SMP PLUS MULTAZAM)**

Disusun oleh:

HASBI KURNIA KARIM

NPM. 301210045

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

**Pada**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TENOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2025

Disetujui oleh:

Penguji 1

Penguji 2

Yusuf Muharam, S.Kom., M.Kom.  
NIK. 04104820003

Yudi Herdiana, S.T., M.T  
NIK. 04104808008

**LEMBAR PENGESAHAN PROGRAM STUDI**

**RANCANG BANGUN APLIKASI PEMBELAJARAN INFORMATIKA  
BERBASIS WEB MENGGUNAKAN ALGORITMA *QUICK SORT*  
(STUDI KASUS SMP PLUS MULTAZAM)**

Disusun oleh:

HASBI KURNIA KARIM

NPM. 3012100045

SKRIPSI ini telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar  
**SARJANA KOMPUTER**

**Pada**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TENOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG**

Baleendah, Agustus 2025

Mengetahui,

Dekan,

Mengesahkan,

Ketua Program Studi,

Yudi Herdiana, S.T., M.T.

NIK. 04104808008

Yusuf Muhamam, S.Kom., M.Kom.

NIK. 04104820003

## **HALAMAN PERNYATAAN**

Yang bertanda tangan dibawah ini, saya:

Nama : Hasbi kurnia Karim

NPM : 301210045

Jurusan : Teknik Informatika

Fakultas : Fakultas Teknologi Informasi

Judul : Rancang Bangun Aplikasi Pembelajaran Informatika Berbasis web

Menggunakan Algoritma Quick sort (Studi Kasus Smp Plus Multazam)

Dalam rangka pengembangan kajian ini, penulis dengan tulus mengungkapkan bahwa penulisan skripsi ini merujuk pada hasil penelitian yang telah dilakukan, serta merupakan hasil pemikiran dan pemaparan yang bersifat orisinal dari penulis sendiri, baik dalam konteks penulisan laporan maupun pelaksanaan aktivitas pemrograman yang dicantumkan sebagai bagian integral dari penelitian ini. Setiap kontribusi atau karya dari pihak lain akan diidentifikasi dengan jelas sesuai dengan pedoman penulisan karya ilmiah yang berlaku, dengan penekanan pada penggunaan referensi yang akurat dan sistematis.

Baleendah, Agustus 2025

Hasbi Kurnia Karim

NPM: 301210045

## ABSTRAK

Perkembangan informasi teknologi mendorong perubahan signifikan dalam sistem pembelajaran modern, khususnya melalui pemanfaatan aplikasi berbasis web yang interaktif yang membuka peluang baru bagi inovasi pedagogis. Penelitian ini merancang dan membangun aplikasi pembelajaran *online* menggunakan metode Waterfall. Aplikasi ini bertujuan tidak hanya meningkatkan efektivitas pembelajaran dan mempermudah interaksi digital antara pengajar dan siswa.

Pengembangan aplikasi pembelajaran berbasis web ini mengadopsi Metode Waterfall berkat alurnya yang sistematis dan terstruktur. Pendekatan ini memastikan setiap tahapan, mulai dari analisis kebutuhan, desain sistem, implementasi, pengujian, hingga pemeliharaan, terlaksana secara berurutan dan terencana. Aplikasi ini dibangun menggunakan PHP sebagai bahasa pemrograman utama dan MySQL untuk basis datanya. Fitur-fitur yang disediakan sangat komprehensif, meliputi materi pembelajaran, latihan soal, forum diskusi, dan evaluasi untuk mendukung proses belajar mengajar yang interaktif. Dirancang dengan prinsip responsif, sistem ini menjamin aksesibilitas yang optimal dari berbagai perangkat, baik desktop maupun mobile, memastikan pengalaman pengguna yang lancar di mana pun Penelitian ini menerapkan studi kasus pada sebuah institusi pendidikan, di mana kebutuhan pengguna diidentifikasi secara mendalam melalui wawancara dan observasi. Hasil pengujian menunjukkan bahwa aplikasi berfungsi dengan baik.

Secara keseluruhan, aplikasi pembelajaran berbasis web ini berhasil diwujudkan dengan pendekatan Waterfall dan telah memenuhi kebutuhan fungsional pengguna. Sistem mendukung proses pembelajaran yang fleksibel, efisien, dan interaktif. Penggunaan teknologi ini menjadi solusi inovatif dalam dunia pendidikan, khususnya di era digital saat ini. Ke depan, pengembangan lebih lanjut dapat mencakup integrasi multimedia dan kecerdasan buatan untuk meningkatkan kualitas pembelajaran secara menyeluruh.

**Kata Kunci:** Aplikasi pembelajaran berbasis web, metode Waterfall algortima *Quick Sort*.

## ***ABSTRACT***

*The development of information technology has driven significant changes in modern learning systems, especially through the use of interactive web-based applications that open up new opportunities for pedagogical innovation. This study designs and builds an online learning application using the Waterfall method. This application aims to not only improve the effectiveness of learning and facilitate digital interaction between teachers and students.*

*The development of this web-based learning application adopts the Waterfall Method thanks to its systematic and structured flow. This approach ensures that each stage, from needs analysis, system design, implementation, testing, to maintenance, is carried out sequentially and in a planned manner. This application is built using PHP as the main programming language and MySQL for its database. The features provided are very comprehensive, including learning materials, practice questions, discussion forums, and evaluations to support an interactive teaching and learning process. Designed with responsive principles, this system ensures optimal accessibility from various devices, both desktop and mobile, ensuring a smooth user experience anywhere. This study applies a case study to an educational institution, where user needs are identified in depth through interviews and observations. The test results show that the application functions well.*

*Overall, this web-based learning application was successfully realized with the Waterfall approach and has met the functional needs of users. The system supports flexible, efficient, and interactive learning processes. The use of this technology is an innovative solution in the world of education, especially in today's digital era. In the future, further developments can include the integration of multimedia and artificial intelligence to improve the quality of learning as a whole.*

***Keywords:*** *Web-based learning application, Waterfall algorithm Quick Sort method.*

## KATA PENGANTAR

Puji Syukur kehadirat Allah SWT, Tuhan Yang Maha Esa, yang telah memberikan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "Rancang bangun aplikasi pembelajaran Informatika berbasis web menggunakan metode waterfall (studi kasus smp plus multazam)". Shalawat serta salam tidak lupa disampaikan kepada junjungan Nabi Muhammad SAW yang senantiasa memberikan teladan dan petunjuk yang luhur dalam setiap aspek kehidupan.

Ucapan terima kasih yang tulus disampaikan kepada kedua Orang Tua yang senantiasa memberikan kasih sayang, dukungan, dan doa yang tidak terbatas. Serta, ucapan terima kasih juga penulis sampaikan kepada pihak yang telah membantu dalam penyusunan proposal skripsi, di antaranya:

1. Bapak Yudi Herdiana, S.T., M.T., selaku Dekan Fakultas Teknologi Informasi Universitas Bale Bandung.
2. Bapak Yusuf Muharam, S.Kom., M.Kom., selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung
3. Rosmalina, S.T., M.Kom. selaku pembimbing utama
4. Segenap Bapak/Ibu Dosen di Fakultas Teknologi Informasi.
5. Semua pihak yangs telah membantu dan memberikan dukungan kepada penulis untuk menyelesaikan usulan penelitian ini.
6. Keluarga ayah, ibu, kaka dan adik, terima kasih semuanya sudah membantu dan mendo'akan memberikan semangat yang telah di berikan selama penyusunan skripsi ini.
7. Rekan- rakan seperjuangan fakultas teknologi informasi angkatan 2021 yang selalu membantu dalam proses penelitian maupun penyesununan skripsi dan menjadi rekan yang luar biasa selama kuliah.
8. Semua pihak yang telah membantu dan memberikan dukungan selama penulisan untuk penyusunuan penelitian ini

Penulis menyadari usulan penelitian ini masih jauh dari sempurna, untuk itu kritik dan saran yang membangun penulis harapkan demi perbaikan dimasa yang akan datang. Akhir kata, penulis berharap semoga proposal ini diterima dan bermanfaat kepada berbagai pihak.

Baleendah, Agustus 2025

Penulis

## DAFTAR ISI

<b>ABSTRAK .....</b>	<b>vi</b>
<i>ABSTRACT .....</i>	<i>vii</i>
<b>KATA PENGANTAR.....</b>	<b>viii</b>
<b>DAFTAR ISI .....</b>	<b>x</b>
<b>DAFTAR GAMBAR .....</b>	<b>xiii</b>
<b>DAFTAR TABEL .....</b>	<b>xv</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>xvi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	4
1.5 Metodologi Penelitian .....	5
1.6 Sistematika Penulisan .....	6
<b>BAB II TINJAUAN PUSTAKA .....</b>	<b>7</b>
2.1 Landasan Teori .....	7
2.2 Dasar Teori .....	11
2.2.1 Aplikasi .....	11
2.2.2 Informatika.....	12
2.2.3 Pembelajaran .....	12
2.2.4 Aplikasi Pembelajaran.....	13
2.2.5 Algoritma <i>Quick Sort</i> .....	13
2.2.6 Visuual Code .....	14
2.2.7 Bahasa Pemrograman PHP .....	14
2.2.8 Database Mysql.....	14
2.2.9 Software Development Life Cycle (SDLC).....	15
2.2.10 Metode Waterfall.....	16
2.2.11 <i>Unified Modeling Language (UML)</i> .....	17
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>24</b>
3.1 Kerangka Pikir .....	24

3.2 Deskripsi .....	25
3.2.1 Identifikasi Masalah .....	25
3.2.2 Pengumpulan Data .....	25
3.2.3 Requirement .....	26
3.2.4 Design .....	27
3.2.5 Implementation .....	30
3.2.6 Verification .....	30
3.2.7 Maintenance .....	30
3.2.8 Pelaporan.....	31
<b>BAB IV ANALISIS DAN PERANCANGAN .....</b>	<b>32</b>
4.1 Analisis.....	32
4.1.1 Analisis Masalah .....	32
4.1.2 Analisis sofware .....	33
4.1.3 Analisis pengguna .....	33
4.1.4 <i>User Interface</i> .....	34
4.1.5 Fitur-Fitur.....	34
4.1.6 Analisis Data .....	36
4.1.7 Analisis Biaya .....	37
4.2 Perancangan .....	37
4.2.1 Perancangan sistem <i>UML</i> .....	38
4.2.2 Perancangan Activity Diagram .....	40
4.2.3 Class daigaram .....	51
4.2.4 Struktut tabel .....	51
4.2.5 Perancangan Desain Antarmuka .....	53
<b>BAB V IMPLEMENTASI DAN PENGUJIAN.....</b>	<b>62</b>
5.1 Implementasi.....	62
5.1.1 Listing program.....	62
5.1.2 Implementasi sistem.....	70
5.1.3 Spesifikasi sistem .....	71
5.1.4 Instalasi sistem .....	72
5.2 Pengujian aplikasi .....	80
5.2.1 black box testing .....	80

<b>BAB VI KESIMPULAN DAN SARAN.....</b>	<b>83</b>
6.1 Kesimpulan .....	83
6.2 Saran.....	84
<b>DAFTAR PUSTAKA.....</b>	<b>85</b>
<b>LAMPIRAN.....</b>	<b>87</b>

## DAFTAR GAMBAR

Gambar 2. 1 Metode SDLC Waterfall.....	16
Gambar 3. 1 Kerangka Pikir.....	24
Gambar 4.1 Use case Diagram User .....	38
Gambar 4. 2 Activity Diagram Login .....	41
Gambar 4. 3 Activity Diagram Dashbroad.....	42
Gambar 4. 4 Activity Diagram Kelas 1 .....	43
Gambar 4. 5 Activity Diagram Kelas 2 .....	44
Gambar 4. 6 Activity Diagram Kelas 3 .....	45
Gambar 4. 7 Activity Diagram Login Admin .....	46
Gambar 4. 8 Activity Diagram Dashbroad Admin.....	47
Gambar 4. 9 Activity Diagram Kelola Akun.....	48
Gambar 4. 10 Activity Diagram Upload Materi Admin.....	49
Gambar 4. 11 Activity Diagram Logout.....	50
Gambar 4. 12 Class Daigram .....	51
Gambar 4. 13 Desain Antarmuka User .....	54
Gambar 4. 14 Buat Akun Baru.....	55
Gambar 4. 15 Desain Antarmuka Dashbroad.....	56
Gambar 4. 16 Desain Antarmuka Materi Kelas 1 .....	56
Gambar 4. 17 Desaim Antarmuka Materi Kelas 2 .....	57
Gambar 4. 18 Desain Antarmuka Materi Kelas 3 .....	58
Gambar 4. 19 Desain Antarmuka Dashbroad Admin.....	58
Gambar 4. 20 Desain Antarmuka Login Admin.....	59
Gambar 4. 21 Desain Antarmuka Kelola User .....	60
Gambar 4. 22 Desain Antarmuka Kelola Materi.....	61
Gambar 51 Halaman login User.....	75
Gambar 5. 2 Halaman Register .....	74
Gambar 5. 3 Halaman Dashboard User.....	75
Gambar 5. 4 Halaman Materi Kelas 1 .....	75
Gambar 5. 5 Halaman Materi Kelas 2.....	76
Gambar 5. 6 Halaman Materi Kelas 3 .....	77

Gambar 5. 7 Halaman Login Admin .....	77
Gambar 5. 8 Dashbroad Admin.....	78
Gambar 5. 9 Kelola User Admin.....	79
Gambar 5. 10 Kelola Materi Admin.....	80

## **DAFTAR TABEL**

Tabel 2. 1 Acuan Jurnal Penelitian.....	7
Tabel 2. 2 <i>Use case Diagram</i> .....	18
Tabel 2. 3 <i>Activity Diagram</i> .....	20
Tabel 2. 4 <i>Sequence diagram Sequence diagram</i> .....	21
Tabel 2. 5 <i>class diagram</i> .....	22
Tabel 4.1 kebutuhan sofware minimun .....	33
Tabel 4.2 Analisis Biaya.....	37
Tabel 4.3 deskripsi aktor .....	39
Tabel 4.4 use case.....	39
Tabel 4.5 tabel user .....	52
Tabel 4.6 tabel siswa .....	52
Tabel 4.7 tabel guru.....	53
Tabel 4.8 tabel materi.....	53
Tabel 5.1 Spesifikasi perangkat keras.....	77
Tabel 5.2 Spesifikasi perangkat lunak.....	78

## **DAFTAR LAMPIRAN**

Lampira 1 hasil wawancara dengan pihak sekolah.....	89
Lampiran 2 Dokumentasi wawancara.....	91
Lampiran 3 Listing program.....	92

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Pembelajaran informatika merupakan upaya sistematis untuk membekali peserta didik dengan pengetahuan dan keterampilan dasar di bidang teknologi informasi dan komunikasi. Dalam konteks perkembangan pendidikan saat ini, informatika tidak hanya mencakup pemahaman terhadap perangkat keras dan lunak, tetapi juga mencakup kemampuan berpikir komputasional, pemrograman, pengolahan data, serta pemecahan masalah berbasis. (Chandrawati et al., 2023).

SMP PLUS MULTAZAM, berlokasi di JL. CINAGGELA NO 73, CINANGGELA, KEC. PACET, KAB. BANDUNG, merupakan sekolah swasta yang berkomitmen untuk mencetak generasi penerus bangsa yang berakhhlak mulia dan berprestasi. Dengan luas tanah mencapai 5.000 m<sup>2</sup>, SMP PLUS MULTAZAM menyediakan lingkungan belajar yang nyaman dan kondusif untuk para siswanya. Sekolah ini menjalankan sistem penyelenggaraan pendidikan pagi selama 6 hari dalam seminggu. Kurikulum yang diterapkan mengacu pada standar nasional dan dipadukan dengan nilai-nilai agama Islam yang kuat.

Berdasarkan hasil wawancara dan observasi di sekolah smp plus multazam. Dalam proses pembelajaran informatika di sekolah smp plus multazam, masih di temukan penerapan penerapan metode konvensional seperti ceramah dan buku cetak yang kurang mampu menumbuhkan minat serta pemahaman siswa terhadap materi. Padahal, mata pelajaran informatika menuntut keterampilan praktis dan pemahaman konseptual yang kuat. Sayangnya, media pembelajaran yang digunakan selama ini masih kurang interaktif dan belum mampu mengakomodasi kebutuhan belajar siswa yang beragam. Selain itu, keterbatasan akses terhadap media pembelajaran yang fleksibel juga menjadi kendala, terutama ketika siswa ingin belajar secara mandiri di luar jam pelajaran. Namun, pemanfaatan teknologi web Belum banyak tersedia di aplikasi pembelajaran informatika berbasis web yang dirancang khusus sesuai dengan kurikulum, karakteristik materi informatika, dan kebutuhan siswa.

Beberapa penelitian terdahulu telah mengembangkan berbagai solusi pembelajaran digital. Pembelajaran digital adalah proses belajar yang memanfaatkan teknologi digital untuk menyampaikan materi ajar, memungkinkan interaksi antara pendidik dan peserta didik melalui perangkat elektronik, serta mendukung fleksibilitas dalam waktu dan tempat belajar.(Chandrawati et al., 2023) Pengembangan e-learning adalah proses merancang dan membangun sebuah sistem pembelajaran berbasis internet yang memungkinkan guru dan siswa berinteraksi di luar batas ruang dan waktu. Sistem ini dirancang untuk mendukung proses belajar mengajar dengan menyediakan akses terhadap materi ajar. (Susanto & Ayu, 2017) Pengembangan e-learning informatika adalah proses merancang dan menciptakan media pembelajaran digital yang interaktif untuk mata pelajaran informatika dengan memanfaatkan teknologi. (Prasetyo & Musril, 2022) Penelitian ini menghadirkan kebaruan dalam bentuk pembuatan aplikasi pembelajaran informatika yang dirancang khusus dengan pendekatan interaktif berbasis web, serta disesuaikan dengan kurikulum yang berlaku. Berbeda dari pembelajaran sebelumnya yang cenderung bersifat monoton dan sulit diakses secara fleksibel, aplikasi ini menawarkan fitur-fitur yang mendukung pembelajaran mandiri, menarik, dan mudah digunakan oleh siswa. Dengan demikian, penelitian ini diharapkan dapat menjadi solusi inovatif yang relevan untuk meningkatkan pemahaman siswa terhadap materi informatika dan Algoritma *Quick Sort* dipilih sebagai fokus materi karena kompleksitasnya yang tinggi dan penting dalam konteks pengurutan data, sehingga dianggap mewakili tantangan utama dalam pembelajaran algoritma. Pemilihan materi ini juga didasarkan pada hasil tinjauan literatur dan identifikasi permasalahan di lapangan.

Sebagai solusi atas berbagai permasalahan dalam pembelajaran informatika, pengembangan aplikasi pembelajaran informatika berbasis web menjadi alternatif yang tepat dan relevan dengan kebutuhan saat ini. Aplikasi ini dirancang untuk menyediakan media pembelajaran yang interaktif, mudah diakses, dan sesuai dengan kurikulum yang berlaku. Dengan memanfaatkan teknologi web, siswa dapat mengakses materi pelajaran informatika secara fleksibel kapan saja dan di mana saja, baik melalui komputer, laptop, maupun perangkat seluler tanpa perlu instalasi aplikasi khusus. Aplikasi ini akan dilengkapi dengan berbagai fitur pendukung

seperti materi pembelajaran informatika, yang dapat meningkatkan keterlibatan serta pemahaman siswa terhadap materi. Di sisi lain, guru juga dapat memanfaatkan aplikasi ini untuk menyampaikan materi dengan cara yang lebih menarik dan efisien. Dalam proses pengembangannya, aplikasi ini akan dirancang menggunakan metode waterfall, yang terdiri dari tahapan analisis kebutuhan, desain sistem, implementasi, pengujian, serta pemeliharaan. Pendekatan ini dipilih karena memberikan struktur kerja yang sistematis dan mudah, sehingga cocok untuk penelitian ini yang memiliki ruang lingkup dan kebutuhan yang telah ditentukan sejak awal. Selain itu, untuk meningkatkan efisiensi pemrosesan data pada aplikasi, akan digunakan algoritma *Quick Sort* dalam pengurutan data tertentu, seperti penyusunan materi berdasarkan kategori atau tingkat kesulitan. Dengan demikian, pengembangan aplikasi pembelajaran informatika berbasis web ini diharapkan mampu menjawab tantangan dalam proses pembelajaran, meningkatkan kualitas pendidikan, dan mendukung transformasi digital di lingkungan sekolah. Berdasarkan latar belakang tersebut, maka penelitian ini berjudul ‘RANCANG BANGUN APLIKASI PEMBELAJARAN INFORMATIKA BERBASIS WEB MENGGUNAKAN ALGORITMA *QUICK SORT* (STUDI KASUS SMP PLUS MULATZAM)

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, terdapat beberapa rumusan masalah yaitu:

1. Bagaimana merancang dan membangun aplikasi pembelajaran informatika berbasis web yang sesuai dengan kebutuhan siswa di SMP Plus Multazam?
2. Bagaimana penerapan algoritma *Quick Sort* dalam sistem aplikasi pembelajaran informatika untuk mendukung pengurutan data seseperti materi bedarsarkan tingkat kesulit, dan sesuai dengan kurikulum sekolah smp plus mulatzam?
3. Sejauh mana aplikasi pembelajaran yang dikembangkan dapat meningkatkan efektivitas dan efisiensi proses belajar mengajar informatika di SMP Plus Multazam?

### 1.3 Batasan Masalah

Adapun batasan masalah terhadap penelitian yang diajukan adalah sebagai berikut:

1. Penelitian ini hanya berfokus pada materi pembelajaran untuk mata pelajaran Informatika di SMP Plus Multazam. Materi yang dikembangkan dibatasi pada kompetensi dasar sesuai Kurikulum yang berlaku di sekolah tersebut, tidak mencakup seluruh mata pelajaran atau jenjang kelas lainnya.
2. Sistem yang dikembangkan terbatas pada fitur utama berupa penyajian modul digital, sistem pengurutan materi menggunakan algoritma *Quick Sort*, dan antarmuka dasar untuk akses konten.
3. Aplikasi dikembangkan khusus untuk platform web dengan teknologi dasar (HTML, CSS, JavaScript) tanpa optimasi khusus untuk perangkat mobile.
4. Evaluasi aplikasi difokuskan pada aspek fungsionalitas, kemudahan penggunaan, dan relevansi materi pembelajaran, bukan pada evaluasi mendalam terhadap hasil belajar secara kuantitatif.
5. Studi kasus di lakukan di smp plus multazam, dengan fokus pada siswa dan beberapa kelas sesuai dengan kebutuhan sekolah.

### 1.4 Tujuan Penelitian

Tujuan yang ingin dicapai penulis pada penelitian yang diajukan adalah sebagai berikut yaitu:

1. merancang dan membangun aplikasi pembelajaran informatika berbasis web yang sesuai dengan kebutuhan pembelajaran siswa di SMP Plus Multazam.
2. Untuk menerapkan algoritma *Quick Sort* dalam sistem aplikasi guna mengelola dan mengurutkan data seperti materi pembelajaran atau nilai siswa secara efisien.
3. Untuk efektivitas dan efisiensi aplikasi pembelajaran yang dikembangkan dalam meningkatkan proses belajar mengajar informatika di SMP Plus Multazam.

## 1.5 Metodologi Penelitian

Dalam proses pengumpulan data, menggunakan beberapa metode agar memperoleh data yang akurat untuk mendukung pembangunan aplikasi dan penyusunan laporan. Adapun metode yang digunakan adalah sebagai berikut:

### A. Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut, yaitu:

#### 1. Observasi

Dalam penelitian skripsi ini, penulis menggunakan beberapa metode pengumpulan data, yaitu observasi langsung kesekolah, wawancara dengan pihak terkait untuk memahami proses kerja, dan studi pustaka dari jurnal serta sumber terkait

#### 2. Wawancara

Dalam penelitian skripsi ini, interaksi langsung antara penulis dan pemilik sekolah melalui percakapan dan tanya jawab. Pengumpulan data dilakukan melalui wawancara yang fokus dan relevan dengan topik penelitian.

#### 3. Studi pustaka

Studi pustaka dilakukan dengan menelaah berbagai literatur yang relevan, seperti jurnal ilmiah, artikel, dan penelitian terdahulu yang membahas pembelajaran informatika atau *e-learning* serta pemanfaatan algoritma *Quick Sort* untuk pengambilan gambar dan pengiriman data. Kajian ini memberikan dasar teori dan teknis untuk mendukung proses perancangan sistem.

### B. Metode pengembangan

Metode pengembangan ini perangkat lunak yang digunakan adalah metode waterfall. Metode waterfall merupakan salah satu pendekatan dalam *Software Development Life Cycle (SDLC)* yang digunakan untuk mengembangkan perangkat lunak. Pendekatan ini mengikuti urutan linear yang dimulai dari tahap analisis, desain, implementasi, pengujian, dan pemeliharaan.

## **1.6 Sistematika Penulisan**

Sistematika penulisan bertujuan untuk menyajikan informasi secara Adapun terstruktur. Adapun sistematika penulisan yang digunakan adalah sebagai berikut:

### **BAB I PENDAHULUAN**

Bab ini mencakup latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian, serta sistematika penulisan.

### **BAB II TINJAUAN PUSTAKA**

Bab ini menjelaskan landasan teori dan konsep-konsep dasar yang berkaitan dengan penelitian, seperti e-learning, pembelajaran informatika, serta metode waterfall. Tinjauan pustaka disusun berdasarkan referensi dari buku, jurnal ilmiah, dan sumber web yang relevan sebagai dasar pengembangan sistem.

### **BAB III METODOLOGI PENELITIAN**

Bab ini menjelaskan metodologi yang digunakan dalam proses perancangan dan pembangunan aplikasi, yaitu metode rekayasa perangkat lunak model waterfall. Tahapan meliputi analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Bab ini juga menjelaskan lingkup penelitian, alat dan bahasa pemrograman yang digunakan.

### **BAB IV ANALISIS, PERANCANGAN, DAN HASIL**

Bab ini membahas tentang metode yang telah ditempuh oleh penulis untuk membuat sistem yang direncanakan berupa rangkaian alat dan fungsi yang tergabung dalam sistem dan rangkaian hasil pengujian sistem.

### **BAB V IMPLEMENTASI DAN PENGUJIAN**

Bab ini berisi mengenai proses penggabungan pada implementasi sistem, spesifikasi sistem, instalasi sistem dan proses menjalankan sistem yang sudah dirancangan dan hasil pengujian pada sistem yang telah dirangkai.

### **BAB VI PENUTUP**

Bab ini berisi kesimpulan dari hasil penelitian dan pengembangan aplikasi yang telah dilakukan, serta saran untuk pengembangan aplikasi di masa yang akan datang

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Landasan Teori

Landasan teori merupakan teori yang memiliki keterkaitan dengan penelitian yang akan dilakukan. Penulis mempelajari penelitian terdahulu sebagai acuan dalam penelitian yang akan dilakukan, hal ini dimaksudkan untuk perbandingan mengenai kesamaan dan perbedaan terhadap penelitian yang terdahulu.

Tabel 2. 1 Acuan Jurnal Penelitian

No	Jurnal acuan	Metode	Pembahasan	Perbedaan
1.	Aplikasi Learning & Test Karyawan dengan Algoritma <i>Quick Sort</i> (Ndaru Ruseno, HS. Sulistiyowati, 2022)	Pemodelan sistem dilakukan dengan menggunakan UML, meliputi Use Case Diagram, Activity Diagram, Sequence Diagram, serta Class Diagram. Proses pengolahan data dalam aplikasi menerapkan algoritma utama <i>Quick Sort.</i>	Pengembangan aplikasi difokuskan pada sistem E-Learning dan E-Test untuk mendukung pelatihan serta proses tes dan seleksi jabatan karyawan. Aplikasi ini dirancang untuk menekan biaya penyelenggaraan tes, memudahkan karyawan dalam mengakses materi pelatihan, serta meningkatkan efektivitas dan	Aplikasi dikembangkan khusus untuk mendukung proses tes dan pelatihan karyawan. Penekanannya terletak pada penghematan biaya serta kemudahan penerapan dalam lingkungan perusahaan.

			efisiensi proses pembelajaran maupun evaluasi.	
2.	Rancang Bangun Aplikasi Tes TOEFL Menggunakan Algoritma <i>Quick Sort</i> Berbasis Komputer (Nuri David Maria Veronika, Yulia Darnita, 2015	Sistem dikembangkan dengan menggunakan pendekatan Prototype, yang mencakup tahap analisis kebutuhan, perancangan, pengujian, hingga implementasi. Aplikasi dibangun dengan bahasa pemrograman VB.Net serta memanfaatkan MS Access sebagai basis data Algoritma utama yang diterapkan adalah <i>Quick Sort</i> untuk proses perangkingan nilai.	Sistem dirancang untuk menyelenggarakan ujian TOEFL secara digital. Algoritma <i>Quick Sort</i> dimanfaatkan untuk mengurutkan hasil nilai peserta sehingga proses perangkingan dapat dilakukan secara otomatis.	Aplikasi dikembangkan berbasis desktop (Windows) Ditujukan untuk konteks akademik dan pendidikan bahasa, bukan untuk kebutuhan perusahaan maupun hiburan/game.
3.	Jurnal Sains dan Teknologi 4.0 (JST 4.0),	Pengumpulan data dilakukan melalui observasi	Algoritma <i>Quick Sort</i> digunakan untuk	Penelitian menitikberatkan pada penerapan

	<p>Vol. 1 No. 2, Juni 2024 Judul: Implementasi Algoritma <i>Quick Sort</i> dalam Sistem Pemesanan Makanan di Rumah Makan</p>	<p>terhadap daftar menu, meliputi informasi nama, harga, serta detail produk.</p> <p>Pengembangan sistem mencakup:</p> <ul style="list-style-type: none"> <li>(1) perumusan kebutuhan,</li> <li>(2) perancangan alur program dengan bahasa pemrograman seperti Python, C++, Java, dan PHP yang terintegrasi dengan MySQL,</li> <li>serta (3) tahap pengujian.</li> </ul> <p>Simulasi algoritma <i>Quick Sort</i> ditunjukkan melalui penggunaan pivot, penyajian flowchart, pseudocode, serta contoh implementasi.</p>	<p>mengurutkan daftar menu dalam sistem pemesanan makanan</p> <p>Penerapan algoritma ini berperan dalam mengurangi potensi kesalahan pencatatan, mempercepat proses transaksi, serta menghasilkan pencatatan pesanan yang lebih terstruktur dan rapi.</p>	<p>nyata algoritma <i>Quick Sort</i> dalam sistem pemesanan, bukan sekadar pada perbandingan antar algoritma</p> <p>Disertakan simulasi, flowchart, pseudocode, serta contoh implementasi program sebagai pendukung Manfaat praktis yang dihasilkan mencakup efisiensi dalam pencatatan, peningkatan kecepatan layanan, ketepatan perhitungan pembayaran, serta meningkatnya kepuasan pelanggan.</p>
--	--	--	---	--

Tiga jurnal yang dijadikan acuan dalam penelitian ini memiliki pendekatan berbeda dalam mengembangkan media pembelajaran digital. Jurnal oleh Desi Rosdiana dan Amri (2024) mengimplementasikan algoritma *Quick Sort* dalam sistem pemesanan makanan di rumah makan. Tujuan penelitian ini adalah mengoptimalkan pengurutan menu agar proses pemesanan lebih cepat, teratur, dan mengurangi potensi kesalahan pencatatan dengan memanfaatkan *Quick Sort*, sistem yang dikembangkan mampu menyusun daftar menu secara efisien sehingga pegawai dapat melayani pelanggan dengan lebih baik dan waktu transaksi menjadi lebih singkat. Jurnal ini membahas pengembangan aplikasi berbasis web dan Android untuk pembelajaran dan pengujian karyawan di PT. Elwilis Mitra Sejahtera menggunakan algoritma *Quick Sort*, dengan tujuan meningkatkan efisiensi pelatihan dan seleksi karyawan yang sebelumnya dilakukan secara manual dan berbiaya tinggi, serta hasilnya berhasil mengurangi biaya operasional dan memungkinkan karyawan mengakses materi dan tes secara fleksibel. Jurnal ini membahas perancangan dan implementasi aplikasi tes TOEFL berbasis komputer menggunakan algoritma *Quick Sort* untuk melakukan perangkingan nilai peserta secara efisien. Aplikasi dibangun dengan bahasa pemrograman VB.Net dan database Microsoft Access, serta menggunakan pendekatan model pengembangan sistem prototipe yang terdiri dari empat tahap: analisis kebutuhan, desain, pengujian, dan implementasi. Hasil penelitian menunjukkan bahwa sebagian besar dari 50 pengguna memperoleh skor antara 300–350 (72%), dan aplikasi ini berhasil memberikan hasil tes serta peringkat secara langsung, meskipun memiliki keterbatasan dalam hal platform (hanya berjalan di Windows) dan akses jaringan.

Penelitian ini memiliki kebaruan dibandingkan dengan tiga jurnal acuan yang telah dikaji. Kebaruan utama terletak pada pengembangan aplikasi pembelajaran informatika berbasis web yang dirancang secara khusus untuk menyajikan materi sesuai kurikulum SMP Plus Multazam, serta menggunakan algoritma *Quick Sort* untuk mengurutkan materi berdasarkan tingkat kesulitan, tanggal pembaruan, dan relevansi. Pendekatan ini belum ditemukan dalam jurnal-jurnal sebelumnya, yang umumnya hanya berfokus pada penyampaian materi secara statis tanpa sistem pengurutan dinamis.

Dari analisis ketiga jurnal tersebut, terdapat adanya gap dalam Penelitian ini dirancang untuk menjawab kebutuhan pembelajaran Informatika yang belum banyak diakomodasi oleh sistem digital yang adaptif dan terstruktur. Aplikasi yang dikembangkan mengintegrasikan algoritma *Quick Sort* sebagai metode pengurutan materi berdasarkan tingkat kesulitan, tanggal pembaruan, dan relevansi, sehingga mempermudah pengguna dalam mengakses konten secara efisien. Selain itu, konten dalam aplikasi disusun secara khusus sesuai dengan kurikulum SMP Plus Multazam, memastikan materi yang disajikan relevan dan terstandarisasi. Penggunaan platform web dengan dukungan framework Laravel memberikan akses tanpa batasan perangkat, serta mendukung proses pembelajaran yang lebih modern, sistematis, dan mudah diakses oleh guru maupun siswa.

## 2.2 Dasar Teori

Adapun teori-teori yang ada sebagai acuan dalam implementasi aplikasi yang akan dibuat yaitu sebagai berikut:

### 2.2.1 Aplikasi

Aplikasi berasal dari bahasa Inggris Application yang artinya penerapan atau penggunaan. Secara teknis back-end aplikasi dibuat oleh para programmer atau developer dengan menggunakan bahasa pemrograman tertentu. Setelah selesai, front-end aplikasi dapat digunakan oleh para *user* dengan menginstalnya pada perangkat elektronik agar dapat digunakan untuk berbagai keperluan, seperti mengolah dokumen, komunikasi, desain grafis, manajemen perangkat keras, bermain game dan lain sebagainya. Aplikasi sendiri terbagi menjadi 3 jenis, yakni, Aplikasi dekstop.

- 1) yang dijalankan pada komputer atau pc. Aplikasi web
- 2) yang dijalankan menggunakan komputer dengan koneksi internet.
- 3) yang dapat dijalankan di perangkat mobile.

Dan pengembangan aplikasi ini ditujukan untuk meningkatkan efektivitas pembelajaran, mengatasi kesulitan belajar mahasiswa, dan meningkatkan motivasi belajar pemrograman.(Hardiyana & Nopandi, 2014)

### **2.2.2 Informatika**

Informatika, sebagai sebuah disiplin ilmu yang fundamental dan terus berkembang, menawarkan pemahaman mendalam terkait pemanfaatan teknologi komputer dalam pengelolaan serta analisis data berukuran besar. Lebih dari sekadar penggunaan perangkat keras dan lunak, informatika memberdayakan kita untuk mengatur dan menginterpretasikan sejumlah besar data dan melalui informasi mesin-mesin berbasis komputasi yang canggih. Keluasan disiplin ilmu ini tercermin dalam beragam bidang yang dicakupnya, meliputi pengembangan dan implementasi sistem informasi yang efektif, landasan teoritis dan algoritmik dalam ilmu komputer, pengelolaan dan organisasi pengetahuan dalam ilmu informasi, perancangan dan implementasi perangkat keras dan perangkat lunak dalam teknik komputer, serta penerapan prinsip-prinsip informatika secara strategis dalam sistem informasi manajemen untuk pengambilan keputusan yang tepat dan efisien. Dalam konteks pendidikan, penerapan informatika menjadi semakin penting, salah satunya melalui pengembangan e-learning mata pelajaran Informatika berbasis Kurikulum Merdeka menggunakan platform Efront di SMP Plus Multazam. Inisiatif ini dirancang untuk mengatasi keterbatasan bahan terbuka dan metode pembelajaran konvensional, sekaligus memberikan solusi digital yang lebih adaptif, interaktif, dan selaras dengan kebutuhan pembelajaran abad ke-21.(Mata et al., 2023)

### **2.2.3 Pembelajaran**

Menurut Undang-Undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional, pendidikan harus menciptakan suasana belajar yang memungkinkan siswa berkembang secara holistik—baik secara spiritual, intelektual, maupun sosial. Dengan kemajuan teknologi yang pesat, media digital dapat dimanfaatkan sebagai alat bantu belajar yang sesuai dengan karakter anak-anak masa kini. Anak-anak sangat mudah tertarik pada perangkat teknologi seperti smartphone dan tablet. Oleh karena itu, pendekatan pembelajaran perlu disesuaikan dengan gaya hidup dan ketertarikan mereka—salah satunya melalui pengembangan aplikasi pembelajaran interaktif.(Tuti, 2022a)

Secara fundamental, proses pendidikan merupakan aktivitas mentransfer pengetahuan dan pemahaman dari pendidik kepada peserta didik. Salah satu teori

pembelajaran, yaitu konstruktivisme, menegaskan bahwa interaksi aktif antara guru dan siswa memiliki peranan krusial dalam keberhasilan pembelajaran. Oleh karena itu, pengembangan sistem e-learning yang mampu mendukung dan memperkuat interaksi tersebut menjadi aspek teoretis yang sangat penting dalam perancangan teknologi pembelajaran.(Bambang Kurnia & Tumini Tumini, 2023)

#### **2.2.4 Aplikasi Pembelajaran**

Pembelajaran berbasis web menjadi salah satu solusi alternatif dalam dunia pendidikan, khususnya saat proses belajar konvensional mengalami hambatan akibat pandemi. Kondisi ini menuntut adanya sistem yang dapat membatasi kehadiran fisik siswa di sekolah sekaligus mencegah penyebaran penyakit. Oleh karena itu, peneliti merancang sebuah aplikasi pembelajaran berbasis web yang dikembangkan menggunakan PHP, Moodle, dan MySQL, dengan tujuan untuk mendukung guru dalam menyampaikan materi secara daring secara efektif selama masa krisis tersebut.(Bambang Kurnia & Tumini Tumini, 2023)

#### **2.2.5 Algoritma *Quick Sort***

Algoritma *Quick Sort*, yang pertama kali diperkenalkan oleh ilmuwan komputer Inggris Tony Hoare pada tahun 1959 dan dipublikasikan pada 1961, dikenal karena efisiensinya dalam manajemen memori serta kecepatannya dalam mengurutkan data dibandingkan metode sorting lainnya. Berkat karakteristik tersebut, *Quick Sort* menjadi algoritma yang banyak digunakan dalam sistem informasi, termasuk pada aplikasi pemesanan makanan, untuk mendukung kinerja sistem yang cepat dan terstruktur.(Rosdiana, 2024)

*Quick Sort* adalah algoritma pengurutan yang bekerja dengan memilih sebuah elemen sebagai pivot, kemudian membandingkannya dengan elemen lainnya dalam daftar. Elemen-elemen yang bernilai lebih kecil dari pivot ditempatkan di sisi kiri, sedangkan elemen yang lebih besar diletakkan di sisi kanan. Proses ini membagi data menjadi dua subdaftar yang masing-masing akan diurutkan kembali secara rekursif, sehingga menghasilkan susunan data yang terstruktur dan efisien.(Kasus et al., 2025)

## 2.2.6 Visuual Code

Visual Studio Code (VS Code) merupakan Integrated Development Environment (IDE) ringan yang dikembangkan oleh Microsoft dan tersedia secara gratis untuk keperluan pribadi maupun komersial. Salah satu kekuatan utama VS Code terletak pada dukungan ekstensinya, yang memungkinkan pengguna menambahkan bahasa pemrograman, debugger, dan berbagai alat bantu lainnya sesuai kebutuhan. Selain ekstensi resmi dari Microsoft, terdapat ribuan ekstensi tambahan yang dikembangkan oleh komunitas dan pengembang independen melalui marketplace resmi. Dalam konteks pendidikan pemrograman dasar, kepraktisan VS Code dianalisis dari empat aspek utama: aksesibilitas, kemudahan penggunaan, kelengkapan fitur, dan popularitasnya di kalangan profesional.(Tan et al., 2024)

## 2.2.7 Bahasa Pemrograman PHP

Aplikasi AjarinCode dibangun menggunakan PHP dengan framework CodeIgniter untuk memfasilitasi pembelajaran bahasa pemrograman web. PHP berperan dalam mengelola logika backend, termasuk pengacakan soal ujian dengan algoritma Fisher-Yates dan penyimpanan data pengguna ke database MySQL. Pemilihan PHP didasarkan pada kemudahan integrasinya dengan teknologi web lain dan dukungan komunitas yang luas(Harry Saptarini et al., 2019)

PHP merupakan salah satu bahasa pemrograman yang paling banyak digunakan saat ini, baik dalam komunitas open source maupun di industri, khususnya untuk membangun aplikasi dan kerangka kerja yang berfokus pada pengembangan web berskala besar.(Siame & Kunda, 2017)

## 2.2.8 Database Mysql

MySQL adalah sistem manajemen database relasional (RDBMS) berbasis open-source yang populer digunakan oleh kebanyakan orang. Umumnya, MySQL digunakan untuk mengelola dan mengakses data dalam jumlah besar dan digunakan oleh untuk mengelola berbagai website maupun aplikasi ternama. MySQL sebagai database untuk menyimpan skor permainan. Database bernama game bangun ruang

memiliki tabel skor yang mencatat nilai integer dari hasil permainan. Skor tertinggi kemudian ditampilkan di halaman khusus. MySQL dipilih karena kemampuannya mendukung pengembangan aplikasi ke versi online di masa depan, meskipun saat ini masih berjalan secara lokal.(Tuti, 2022b)

MySQL merupakan sistem manajemen basis data relasional berbasis arsitektur client-server yang dikenal luas karena kinerjanya yang tinggi, keandalannya, serta kemampuannya untuk diskalakan. Untuk mencapai performa optimal, diperlukan konfigurasi sistem yang tepat serta pengoptimalan penyimpanan fisik data. Strategi seperti pemrograman fisik dan optimasi data terbukti mampu meningkatkan efisiensi dan skalabilitas sistem database MySQL. Selain itu, penerapan indeks yang tepat, normalisasi yang seimbang, serta pengaturan cache dan buffer yang optimal juga menjadi faktor penting dalam menunjang kinerja MySQL dalam lingkungan produksi yang kompleks dan dinamis..(Šušter & Ranisavljević, 2023)

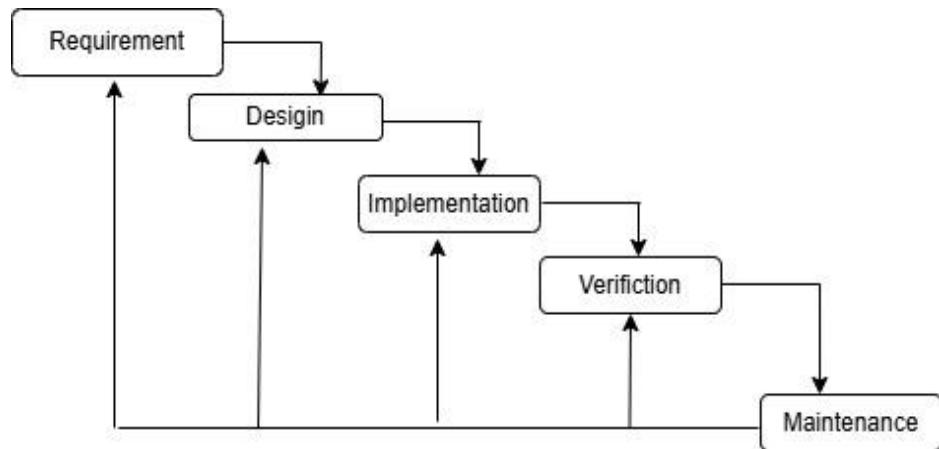
### **2.2.9 Software Development Life Cycle (SDLC)**

Penelitian ini menggunakan metode System Development Life Cycle (SDLC) dalam pengembangan aplikasi pembelajaran informatika. Proses pengembangan mencakup beberapa tahapan, yaitu: identifikasi kebutuhan, analisis sistem, perancangan antarmuka pengguna, implementasi menggunakan App Inventor, serta pengujian fungsionalitas aplikasi. Hasil akhir menunjukkan bahwa aplikasi berhasil dibangun dengan fitur utama berupa materi ajar, yang dilengkapi dengan tampilan skor dan durasi penggeraan. Pemilihan metode SDLC didasarkan pada kemampuannya dalam memberikan struktur yang sistematis dan terarah untuk penelitian ini dengan tujuan pengembangan perangkat lunak yang spesifik. Perancangan kebutuhan sistem dilakukan dengan merumuskan spesifikasi kebutuhan sistem dan menyusun perencanaan yang matang sebagai dasar pelaksanaan seluruh tahapan dalam penelitian ini.

- 1) Konstruksi sistem - merancang alur kerja manajemen dan desain pemrograman untuk pengembangan sistem.
- 2) Implementasi sistem - menerapkan sistem sesuai dengan fungsionalitas yang telah ditetapkan.

- 3) Pengujian sistem - melakukan evaluasi terhadap sistem yang telah dibangun; serta
- 4) Pemeliharaan sistem - mengoperasikan dan merawat sistem yang telah diimplementasikan"(Mukti et al., 2016)

### 2.2.10 Metode Waterfall



Gambar 2. 1 Metode SDLC Waterfall

(sumber Aceng 2021)

Model waterfall pertama kali diperkenalkan oleh *Winston Royce* sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai didalam Software Engineering (SE). saat ini model waterfall merupakan model pengembangan perangkat lunak yang sering digunakan.

#### 1. Requirement

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

#### 2. Design

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras (hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

### 3. Implementation

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit testing.

### 4. Verification

Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit testing (dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem beraksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan atau nama pelanggan untuk melihat apakah semua kebutuhan pelanggan puas).

### 5. Maintenance

Ini adalah tahap akhir dari metode waterfall. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

#### **2.2.11 Unified Modeling Language (UML)**

*Unified Modeling Language (UML)* adalah standar bahasa yang digunakan untuk memodelkan, mendokumentasikan, dan mengembangkan perangkat lunak. *UML* merupakan metode dalam pengembangan sistem berbasis objek yang berperan sebagai alat bantu dalam proses perancangan. Beberapa diagram yang digunakan dalam pemodelan berbasis objek dengan *UML* antara lain *Use case Diagram*, *Activity Diagram*, *sequence diagram*, dan *class diagram*.

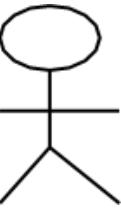
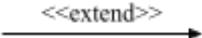
Berikut adalah diagram-diagram dalam *UML*:

##### 1. Use case Diagram

*Diagram use case* pertama kali diperkenalkan oleh Ivar Jacobson pada tahun 1986. Diagram ini merupakan pendekatan dalam analisis sistem yang bertujuan untuk mengidentifikasi, menjelaskan, serta mengorganisir kebutuhan sistem. Dalam *UML (Unified Modeling Language)*—yang merupakan notasi standar untuk memodelkan objek dan sistem dunia nyata—*diagram use case* termasuk dalam kategori diagram perilaku. Diagram ini merupakan salah satu jenis diagram berorientasi objek yang menggambarkan interaksi antara sistem dengan entitas.

Tabel 2. 2 *Use case Diagram.*

(sumber Arwa 2016)

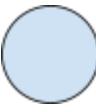
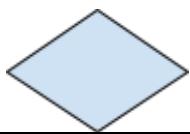
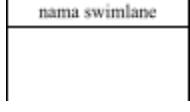
No	Simbol	Keterangan
1.	<i>Use Case</i> 	Fungsionalitas yang disediakan sistem sebagai unit yang bertukar pesan antar unit atau aktor
2.	<i>Aktor/actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>Aktor</i> adalah gambar orang, tapi <i>Aktor</i> belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawali nama aktor.
3.	<i>Asosiasi/association</i> 	Komunikasi antar <i>Aktor</i> dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	<i>Ekstensi/extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> , yang dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri tanpa <i>use case</i> tambahan tersebut
5.	<i>Generalisasi/generalization</i> 	Hubungan generalisasi dan spesialisasi antara dua buah <i>use case</i> , dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
6.	<i>Menggunakan / Include / uses</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan memerlukan

		<i>use</i>
7.		<p>case ini sebagai syarat untuk menjalankan fungsinya.</p> <p>Ada dua sudut pandang yang cukup besar mengenai include di use case:</p> <ul style="list-style-type: none"> <li>a. Include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan</li> <li>b. Include berarti use case yang ditambahkan akan selalu melakukan pengecekan apakah use case yang ditambahkan telah dijalankan sebelum use case tambahan dijalankan</li> </ul>

## 2. Activity Diagram

*Diagram aktivitas UML* merupakan jenis diagram perilaku yang menggambarkan perilaku internal dari berbagai operasi dalam sebuah program dengan memanfaatkan simpul (nodes) dan sisi (edges). Diagram ini banyak digunakan dalam berbagai bidang untuk merepresentasikan alur kerja. Konsep dasar diagram aktivitas diadaptasi dari Petri-net, diagram alir (*flowchart*), dan diagram transisi status, namun memiliki kelebihan berupa dukungan terhadap aktivitas paralel, perulangan, serta perilaku yang dipicu oleh kejadian. *Diagram Aktivitas UML* (UAD) dimanfaatkan untuk memodelkan berbagai elemen seperti aktivitas, sub-aktivitas, transisi, titik keputusan, kondisi penjaga, aktivitas paralel, percabangan, penggabungan, *swim lane*, serta penggabungan dan percabangan proses. Setiap *diagram aktivitas* dimulai dengan satu aktivitas awal dan berakhir dengan satu aktivitas akhir. (Muhammad Touseef Ikram 2015)

Tabel 2. 3 *Activity Diagram*  
(sumber Muhammad Touseef Ikram 2015)

No.	Simbol	Keterangan
1.	Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2.	Aktivitas 	Aktivitas yang dilakukan sistem, biasanya diawali dengan kata kerja.
3.	Percabangan/ <i>Decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan/ <i>Join</i> 	Asosiasi penggabungan/ <i>join</i> dimana lebih dari satu aktivitas digabungkan menjadi satu
5.	Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

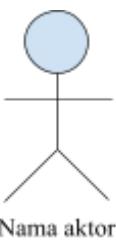
### 3. *Sequence diagram*

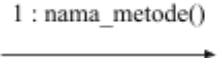
Diagram sekuens terutama digunakan untuk mengilustrasikan bagaimana objek berinteraksi satu sama lain dalam urutan terjadinya interaksi tersebut. Meskipun sering dianggap sebagai alat yang khusus untuk pengembang, staf bisnis

organisasi juga dapat menganggapnya berharga untuk mengomunikasikan proses bisnis saat ini dengan menunjukkan interaksi objek.

Tabel 2. 4 *Sequence diagram Sequence diagram*

(Sumber IBM 2005)

NO.	Simbol	Keterangan
1.	 Aktor Nama aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem sistem informasi yang akan dibuat itu sendiri.
2.	 Garis hidup/lifeline	Menyatakan kehidupan suatu objek.
3.	 Objek <u>Nama objek: nama kelas</u>	Menyatakan objek yang berinteraksi.
4.	 Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif adalah sebuah tahapan yang dilakukan di dalamnya.
5.	 Pesan tipe create	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
6.	Pesan tipe call	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain

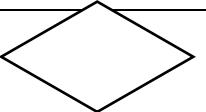
		atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi
7.		Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
8.		Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian.

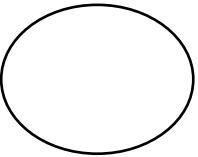
#### 4. Class Diagram

Diagram kelas digunakan untuk menggambarkan struktur statis dari suatu sistem yang sedang dimodelkan. Diagram ini menampilkan elemen-elemen dalam sistem beserta struktur internal dan hubungan antar elemen tersebut. Karena hanya mencerminkan struktur statis, diagram kelas menyajikan jenis-jenis objek atau entitas yang ada dalam sistem, bukan representasi instansinya secara spesifik.(DAUM, 2003)

Tabel 2. 5 *Class Diagram*

(sumber Daum 2003)

No	Simbol	Keterangan
1.		Hubungan dimana objek anak ( <i>descendent</i> ) berbagai prilaku dan struktur data dari sebagai objek induk.
2.		upaya untuk menghindari asosiasi dengan lebih dari 2 objek.

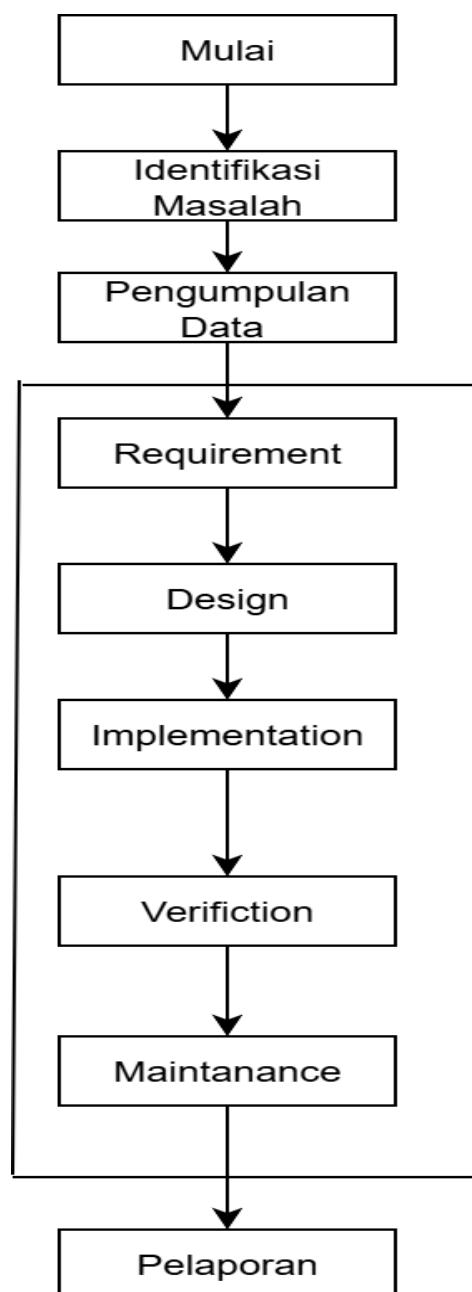
3.		Operasi yang benar benar di lakukan oleh suatu objek
4.		Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
5.		Apa yang menghubungkan antara objek satu dengan yang lainya.
6.		Deskripsi dari urutan aksi- aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi aktor.

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Kerangka Pikir**

Kerangka pikir merupakan jalur pemikiran yang dirancang berdasarkan kegiatan penyusun yang dilakukan. Berikut adalah kerangka pikir yang merupakan langkah-langkah yang akan dilakukan dalam penelitian ini.



Gambar 3. 1 Kerangka Pikir

## 3.2 Deskripsi

Deskripsi ini merupakan penjabaran dari kerangka berpikir yang telah dirancang dalam penelitian.

### 3.2.1 Identifikasi Masalah

Pada tahap ini dilakukan pengumpulan informasi awal untuk memahami secara menyeluruh permasalahan yang dihadapi oleh siswa dan guru di SMP Plus Multazam dalam proses pembelajaran mata pelajaran Informatika. Proses ini dilaksanakan melalui observasi langsung terhadap kegiatan belajar mengajar di kelas, serta wawancara dengan guru dan pihak sekolah untuk mengidentifikasi hambatan yang kerap muncul. Berdasarkan hasil pengumpulan data, ditemukan sejumlah permasalahan utama yang menjadi dasar dari pengembangan aplikasi ini, di antaranya: keterbatasan media pembelajaran digital yang interaktif, ketergantungan pada metode ceramah dan buku cetak yang kurang menarik minat siswa, serta belum tersedianya sistem terstruktur untuk penyampaian materi yang sesuai dengan kurikulum. Selain itu, guru juga mengalami kesulitan dalam menyusun dan menyampaikan materi secara sistematis, serta dalam memantau perkembangan belajar siswa secara digital. Permasalahan ini berdampak pada rendahnya efektivitas proses pembelajaran dan kurangnya keterlibatan siswa dalam memahami konsep-konsep dasar Informatika. Oleh karena itu, perlu dikembangkan sebuah aplikasi pembelajaran berbasis web yang dirancang secara khusus untuk menyajikan materi Informatika secara interaktif, terstruktur, dan mudah diakses, dengan didukung oleh algoritma *Quick Sort* guna mengelola penyusunan materi berdasarkan tingkat kesulitan.

### 3.2.2 Pengumpulan Data

#### 1. Observasi

Metode ini melibatkan pengamatan langsung ke sekolah dan mendapatkan informasi terperinci tentang sekolah Informasi yang diperoleh kemudian digunakan sebagai dasar untuk mengaplikasikan solusi yang sesuai. Observasi dilakukan dengan mengorganisir pemantauan dan pencatatan

langsung terhadap aktivitas di sekolah. Pendekatan ini memungkinkan untuk memahami praktik yang sebenarnya berlangsung dan menganalisis aspek-aspek yang relevan dalam pembelajaran.

## 2. Wawancara

Metode ini dilakukan dengan interaksi langsung antara penulis dan pemilik sekolah melalui percakapan dan tanya jawab. Pengumpulan data dilakukan melalui wawancara yang fokus dan relevan dengan topik penelitian, yaitu rancang bangun aplikasi pembelajaran. Dalam proses wawancara, informasi yang diperlukan tentang sistem pemesanan dapat dikumpulkan langsung dari pihak yang terlibat, seperti pemilik atau staf sekolah. Pendekatan ini memungkinkan untuk mendapatkan pemahaman yang lebih dalam tentang bagaimana sistem sekolah saat ini berfungsi dan mendapatkan masukan langsung dari pemangku kepentingan.

## 3. Studi Pustaka

Pengumpulan data pada penelitian ini berkaitan dengan analisis dari berbagai tulisan dalam jurnal ilmiah dan buku yang sudah terbit dan berkaitan dengan topik penelitian yang penulis ambil. Teknik pada pengumpulan data dilakukan dengan membaca buku-buku, referensi dan jurnal-jurnal. Pada penelitian ini penulis menggunakan beberapa jurnal dengan menggunakan metode algoritma *Quick Sort* untuk mengetahui seberapa efektif dan akurat aplikasi pembelajaran tersebut.

### 3.2.3 Requirement

Pada tahap ini dilakukan proses analisis secara menyeluruh untuk memahami kebutuhan, hambatan, serta peluang dalam pengembangan aplikasi pembelajaran informatika berbasis web di SMP Plus Multazam. Proses analisis mencakup pengumpulan data melalui observasi terhadap proses pembelajaran yang sedang berjalan, serta wawancara dengan guru dan siswa untuk menggali kebutuhan yang belum terpenuhi oleh metode konvensional. Berdasarkan data yang diperoleh, dilakukan identifikasi terhadap kekurangan sistem pembelajaran saat ini, seperti kurangnya media interaktif, keterbatasan akses materi di luar jam pelajaran, serta kesulitan dalam menyusun materi sesuai kurikulum secara sistematis. Selain itu,

dilakukan analisis kebutuhan perangkat lunak dan perangkat keras yang dibutuhkan, termasuk penentuan fitur utama aplikasi, struktur antarmuka pengguna, dan peran algoritma *Quick Sort* dalam mengelola penyusunan materi berdasarkan tingkat kesulitan. Hasil analisis ini menjadi dasar penting dalam merancang arsitektur sistem dan memastikan bahwa aplikasi yang dikembangkan mampu menjawab kebutuhan pembelajaran informatika secara efektif dan efisien.

### 3.2.4 Design

Desain sistem yang digunakan dalam penelitian ini meliputi beberapa tahapan, yaitu:

1) Perancangan Diagram

Pada tahap ini dilakukan dengan membuat desain menggunakan bahasa pemodelan *Unified Modelling Language (UML)*. *UML* yang digunakan dalam penelitian ini terdiri dari:

a. *Use case Diagram*

*Use case diagram* adalah alat visual untuk menggambarkan hubungan antara aktor dan kasus penggunaan dalam sebuah sistem aplikasi. Diagram ini membantu memahami interaksi antara aktor, yang berinteraksi dengan sistem, dan fungsi-fungsi yang ada dalam sistem. Dalam penelitian ini, terdapat dua aktor utama: "Admin" dan "siswa", serta kasus penggunaan yang meliputi:

1. *Login admin*: Memberikan akses kepada admin agar dapat masuk ke dalam sistem aplikasi dengan menggunakan kredensial yang valid. Hal ini memastikan bahwa hanya admin yang berwenang yang dapat mengakses dan mengelola fungsi-fungsi dalam aplikasi.
2. *Login siswa*: siswa dapat masuk ke dalam aplikasi pembelajaran menggunakan akun masing-masing. Setelah *Login*, siswa dapat mengakses materi pelajaran, mengikuti kelas, mengerjakan tugas. Fitur ini dirancang untuk mendukung proses belajar yang mandiri, terstruktur, dan mudah diakses kapan saja.
3. *Logout*: Mengakhiri sesi pengguna dari sistem dan memastikan akun keluar dengan aman.

4. Kelola akses materi admin: admin untuk menambahkan, mengedit, atau menghapus materi pembelajaran yang tersedia di sistem.
5. Kelola user admin: Admin dapat mengelola data pengguna seperti menambahkan akun siswa, mengatur hak akses, dan menghapus user jika diperlukan.
6. Halaman dashboard user: Halaman ini menjadi pusat navigasi bagi siswa setelah login, menampilkan informasi dan akses ke materi ajar
7. Halaman materi ajar kelas 1: Berisi kumpulan materi dasar informatika yang ditujukan untuk siswa kelas 1. Materi disusun secara bertahap untuk membangun pemahaman awal siswa terhadap konsep informatika.
8. Halaman materi ajar kelas 2: Menampilkan materi lanjutan yang lebih kompleks, sesuai dengan tingkat perkembangan siswa kelas 2. Kontennya bertujuan memperluas wawasan logika dan keterampilan komputasional.
9. Halaman materi ajar kelas 3: Halaman ini menyediakan materi tingkat lanjut, termasuk algoritma. Tujuannya adalah melatih kemampuan analisis dan berpikir kritis siswa sebelum mereka lulus.

b. *Activity diagram*

*Activity diagram* digunakan untuk memodelkan rangkaian aktivitas dalam *suatu* sistem, mencakup representasi visual dari urutan aktivitas yang dilakukan oleh pengguna atau entitas lain dalam aplikasi. Diagram ini memberikan pandangan keseluruhan tentang bagaimana aktivitas berlangsung, berinteraksi, dan mengalir dalam sistem. Dalam konteks penelitian ini, penulis akan membuat beberapa diagram aktivitas yang berkaitan dengan sistem aplikasi. Di antaranya:

1. Activity diagram login user: Menggambarkan proses siswa saat melakukan login ke dalam sistem menggunakan akun yang telah terdaftar untuk mengakses materi pembelajaran.
2. Activity diagram dashboard: Menggambarkan aktivitas siswa setelah berhasil login, yaitu mengakses halaman dashboard yang menampilkan daftar materi pembelajaran berdasarkan kelas.
3. Activity diagram materi ajar kelas 1: Menjelaskan alur siswa saat membuka materi pembelajaran kelas 1 yang berisi pengenalan dasar-dasar informatika.

4. Activity diagram materi ajar kelas 2: Menggambarkan proses siswa dalam mengakses materi pembelajaran tingkat lanjut untuk kelas 2, seperti struktur data dan logika pemrograman.
5. Activity diagram materi ajar kelas 3: Menunjukkan alur aktivitas siswa ketika membuka materi pembelajaran kelas 3 yang membahas algoritma.
6. Activity diagram login admin: Menggambarkan alur aktivitas admin saat masuk ke dalam sistem dengan memasukkan email dan kata sandi yang valid.
7. Activity diagram dashboard admin: Menggambarkan alur aktivitas admin setelah berhasil login, yaitu mengakses halaman utama yang menampilkan menu pengelolaan materi.
8. Activity diagram kelola user admin: Menjelaskan alur aktivitas admin dalam mengelola data pengguna, termasuk menambahkan akun siswa baru dan memperbarui informasi pengguna yang sudah ada.
9. Activity diagram kelola materi admin: Menunjukkan langkah-langkah yang dilakukan admin dalam menambah, mengedit, atau menghapus materi pembelajaran sesuai dengan kurikulum.
10. Activity diagram logout: Menjelaskan proses saat pengguna (admin atau siswa) keluar dari sistem untuk mengakhiri sesi dan menjaga keamanan akun.

c. *Class Diagram*

*Class diagram* untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class Diagram* yang akan dibuat adalah sebagai berikut:

- 1) *Class Diagram* admin
  - 2) *Class Diagram* pengelola pembelajaran
  - 3) *Class Diagram* user
- 2) Perancangan Database

Perancangan database pada sistem ini menggunakan MySQL yang berbasis tabel. Setiap data penting, seperti pengguna (admin dan siswa), materi pembelajaran, dan aktivitas pengguna, disimpan dalam tabel yang berbeda. Tabel-tabel ini saling terhubung melalui kolom yang sama, seperti ID pengguna atau ID materi. Dengan cara ini, data menjadi lebih rapi, mudah dicari, dan dikelola. Selain

itu, database ini juga dirancang agar dapat menampilkan materi sesuai tingkat kesulitan dengan bantuan algoritma *Quick Sort*.

### 3) Perancangan Antarmuka

Pada tahapan perancangan antarmuka, desain aplikasi dibuat sederhana menggunakan figma, dapat mengilustrasikan aplikasi yang dibangun dan tentunya memperhatikan interaksi yang akan terjadi dalam aplikasi antara *user* dan sistem. Selain aspek visual, desain antarmuka juga memperhatikan interaksi yang akan terjadi antara pengguna dan sistem.

#### 3.2.5 Implementation

Tahap implementasi aplikasi pembelajaran informatika berbasis web dilakukan menggunakan PHP sebagai bahasa pemrograman utama dengan dukungan MySQL sebagai basis data untuk mengelola informasi pengguna, materi ajar, serta data pendukung lainnya secara terstruktur. PHP digunakan untuk membangun logika backend dan mengatur alur kerja aplikasi, sedangkan HTML, CSS, dan JavaScript dimanfaatkan untuk membentuk antarmuka pengguna yang responsif dan mudah digunakan.

#### 3.2.6 Verification

Setelah tahap implementasi selesai dilaksanakan, proses selanjutnya adalah melakukan pengujian sistem menggunakan metode Black Box Testing untuk memastikan bahwa setiap fitur dan fungsionalitas yang terdapat dalam aplikasi pembelajaran informatika berbasis web berjalan sesuai dengan rancangan yang telah ditetapkan pada tahap desain. Metode ini difokuskan pada pengujian perilaku sistem dari sisi pengguna tanpa melihat kode program secara langsung.

#### 3.2.7 Maintenance

sistem yang telah diimplementasikan akan dijalankan dan dirawat untuk memastikan stabilitas serta kesesuaian dengan kebutuhan pengguna. Tahap ini mencakup perbaikan bug yang tidak ditemukan sebelumnya, pembaruan materi

ajar, peningkatan fitur berdasarkan umpan balik, serta menjaga performa sistem agar tetap responsif dan relevan.

### **3.2.8 Pelaporan**

Tahap akhir dari penelitian ini adalah penyusunan laporan, yang menjadi langkah penting dalam menyelesaikan tugas. Laporan ini disusun mengikuti pedoman Penulisan Laporan Skripsi Fakultas Teknologi Informasi Universitas Bandung, dengan memastikan kesesuaian dengan standar akademik yang berlaku serta kepatuhan terhadap peraturan yang ditetapkan. Penyusunan laporan ini mencakup analisis, implementasi, serta hasil pengujian sistem yang telah dibangun, yang akan menjadi bukti keberhasilan dalam menyelesaikan penelitian ini.

## **BAB IV**

### **ANALISIS DAN PERANCANGAN**

#### **4.1 Analisis**

Analisis dilakukan sebagai langkah awal penelitian untuk mengidentifikasi kebutuhan-kebutuhan yang diperlukan dalam merancang aplikasi. Berikut adalah analisis yang telah penulis lakukan untuk membantu dalam perancangan aplikasi pembelajaran informatika. Analisis ini mencakup pembahasan mengenai analisis masalah, analisis software, analisis pengguna, *user interface*, fitur-fitur, dan analisis biaya.

##### **4.1.1 Analisis Masalah**

Langkah awal dalam penelitian ini adalah melakukan analisis terhadap sistem pembelajaran informatika yang sedang berjalan di SMP Plus Multazam. Tujuan dari analisis ini adalah untuk memahami metode pengajaran yang digunakan saat ini, menilai efektivitas media pembelajaran konvensional, serta mengidentifikasi permasalahan yang dihadapi oleh siswa dan guru dalam proses belajar-mengajar. Hasil dari analisis ini menjadi dasar dalam perancangan aplikasi pembelajaran berbasis web yang lebih interaktif dan adaptif, sesuai dengan kebutuhan kurikulum dan karakteristik siswa di sekolah tersebut.

Berdasarkan hasil observasi dan wawancara dengan pihak sekolah, ditemukan beberapa permasalahan utama. Pertama, proses pembelajaran masih didominasi oleh metode ceramah dan penggunaan buku cetak, yang menyebabkan rendahnya minat belajar siswa serta kurangnya pemahaman terhadap konsep-konsep dasar informatika yang bersifat praktikal. Kedua, materi ajar yang digunakan belum terstandar secara sistematis dan sering kali diambil dari berbagai sumber eksternal yang tidak sepenuhnya sesuai dengan kurikulum sekolah.

Selain itu, keterbatasan akses terhadap media pembelajaran digital menyebabkan siswa tidak dapat belajar di luar jam pelajaran formal. Belum adanya sistem pembelajaran online guru dalam mengelola materi ajar secara efisien dan memantau perkembangan siswa secara menyeluruh.

Permasalahan-permasalahan tersebut menunjukkan bahwa sistem pembelajaran yang ada belum optimal dan belum mendukung kebutuhan pembelajaran di era digital saat ini. Oleh karena itu, pembuatan aplikasi pembelajaran berbasis web dengan algoritma *Quick Sort* diusulkan sebagai solusi untuk meningkatkan efektivitas proses belajar mengajar. Aplikasi ini diharapkan mampu memberikan kemudahan akses, penyajian materi yang terstruktur berdasarkan tingkat kesulitan, serta antarmuka yang mendukung pembelajaran mandiri dan interaktif.

#### **4.1.2 Analisis sofware**

Berdasarkan analisis kebutuhan, untuk memenuhi persyaratan dalam pembuatan aplikasi pembelajaran informatika diperlukan beberapa perangkat lunak sebagai alat bantu. Perangkat lunak yang dibutuhkan dalam penelitian ini adalah sebagai berikut:

Tabel 4.1 kebutuhan sofware minimun

Sistem operasi	Windows 10
Code editor	Viusal studio code
Database	MYSQL
Perancangan user interface	Figma
Perancangan UML	Draw.io

#### **4.1.3 Analisis pengguna**

Dalam pengembangan aplikasi pembelajaran informatika berbasis web di SMP Plus Multazam, pengguna sistem terdiri dari dua peran utama, yaitu admin (guru) dan *user* (siswa). Admin, yang merupakan guru itu sendiri, memiliki peran sebagai pengelola utama sistem, mulai dari membuat, mengedit, hingga menghapus materi pembelajaran yang akan digunakan oleh siswa. Selain itu, admin juga bertanggung jawab dalam mengelola struktur konten, mengatur tingkat kesulitan

materi, serta memastikan materi yang tersedia sesuai dengan kurikulum. Sementara itu, *user* adalah siswa yang menggunakan aplikasi untuk mengakses materi pelajaran, belajar secara mandiri, dan mengikuti alur pembelajaran yang telah disediakan. Setiap jenis pengguna memiliki tampilan dan hak akses yang berbeda, yang disesuaikan dengan kebutuhan masing-masing peran untuk memastikan efisiensi, kemudahan, dan efektivitas dalam proses belajar mengajar berbasis web.

#### **4.1.4 User Interface**

*User Interface (UI)* pada aplikasi pembelajaran informatika berbasis web ini dirancang dengan tampilan yang sederhana, mudah dipahami, dan hanya difokuskan untuk akses melalui perangkat desktop atau laptop. Terdapat dua jenis pengguna utama, yaitu admin/guru dan siswa, yang masing-masing memiliki tampilan antarmuka yang disesuaikan dengan peran mereka. Setelah *login*, siswa akan diarahkan ke halaman dashboard yang menampilkan daftar materi pembelajaran berdasarkan kelas dan tingkat kesulitan yang telah diurutkan menggunakan algoritma *Quick Sort*. Sementara itu, admin yang merupakan guru memiliki tampilan antarmuka serupa, namun dengan akses tambahan untuk mengelola konten pembelajaran, seperti menambahkan, mengedit, dan menghapus materi ajar. Seluruh elemen antarmuka didesain agar memudahkan navigasi dan mendukung kelancaran proses belajar mengajar secara digital di lingkungan sekolah.

#### **4.1.5 Fitur-Fitur**

Fitur-fitur yang tersedia dalam aplikasi pembelajaran ini dirancang untuk memberikan kemudahan kepada siswa, guru, dan admin dalam menjalankan proses belajar mengajar secara digital. Dengan menggunakan aplikasi ini, pengguna dapat dengan mudah mengakses materi pelajaran, berinteraksi secara online, dan melakukan evaluasi pembelajaran dengan lebih efisien. Berikut adalah fitur-fitur yang disediakan:

## 1. Halaman *Login*

Halaman ini merupakan akses awal ke dalam aplikasi, di mana pengguna (admin, dan siswa) harus memasukkan username dan *password*. Sistem akan memverifikasi data login untuk memberikan akses yang sesuai dengan pengguna.

## 2. Halaman Dashboard

Setelah berhasil login, pengguna akan langsung diarahkan ke dashboard. Di sinilah seluruh informasi utama dan navigasi penting ditampilkan, seperti pilihan kelas, aktivitas pengguna. Dashboard disesuaikan berdasarkan jenis pengguna: siswa dapat melihat daftar materi yang tersedia, guru dapat mengelola konten pembelajaran.

## 3. Halaman Pembelajaran Kelas 1

Menampilkan materi dasar informatika untuk siswa kelas 1. Materi mencakup pengenalan komputer dan dasar-dasar logika algoritma. Guru dapat menambahkan atau memodifikasi materi sesuai kurikulum, sedangkan siswa dapat mengakses dan mempelajarinya secara mandiri.

## 4. Halaman Pembelajaran Kelas 2

Halaman ini menyajikan materi lanjutan seperti struktur data dan pengantar pemrograman. Konten didesain untuk memperluas wawasan setelah siswa memahami materi dasar, dengan pendekatan yang interaktif untuk meningkatkan keterlibatan siswa.

## 5. Halaman Pembelajaran Kelas 3

Materi tingkat lanjut tersedia di halaman ini, fokus pada algoritma kompleks seperti *Quick Sort*. Diperuntukkan bagi siswa yang sudah menguasai materi sebelumnya, fitur ini bertujuan untuk mengembangkan kemampuan analisis dan berpikir komputasional siswa.

## 6. Halaman *Logout*

Digunakan untuk keluar dari aplikasi dengan aman setelah pengguna menyelesaikan aktivitasnya. Ini penting untuk menjaga keamanan akun dan memastikan data pengguna tetap terlindungi dari akses yang tidak sah.

#### 4.1.6 Analisis Data

Untuk membangun aplikasi pembelajaran informatika menggunakan algoritma *Quick Sort* penulis memerlukan sejumlah data yang akan digunakan dan diolah dalam aplikasi tersebut. Berikut adalah beberapa data yang akan digunakan dalam penelitian dan perancangan aplikasi:

1. Data siswa

Data siswa merupakan komponen penting dalam sistem aplikasi pembelajaran informatika, karena menjadi dasar untuk mengelola akses, aktivitas belajar, dan evaluasi pengguna. Data ini dikumpulkan saat siswa mendaftar atau didaftarkan ke dalam sistem oleh admin atau guru, dan digunakan untuk menyusun pengalaman belajar yang lebih personal dan terstruktur.

2. Data guru

dalam aplikasi berfungsi sebagai dasar untuk mengelola peran guru dalam proses pembelajaran digital, khususnya dalam menyusun materi, memantau siswa, dan melakukan evaluasi. Data ini dikumpulkan saat guru didaftarkan ke dalam sistem oleh admin, dan digunakan untuk memastikan akses yang aman serta fungsionalitas sistem sesuai dengan peran guru.

3. Data pembelajaran

Pembelajaran merupakan inti dari sistem aplikasi, berisi informasi mengenai materi-materi informatika yang disediakan untuk siswa. Data ini dikelola oleh guru dan disusun secara terstruktur berdasarkan kategori, topik, dan tingkat kesulitan. Sistem juga menggunakan algoritma *Quick Sort* untuk mengurutkan materi pembelajaran agar mudah diakses sesuai kebutuhan siswa.

Tabel 4.2 Implementasi Algoritma *Quick Sort*

No.	Materi	Kelas	Tingkat kesulitan
1.	bk	7	Mudah
2.	Pengolahan data	7	mudah
3.	Literasi informasi	7	Mudah
4.	Kesimbangan hidup di dunia di gital	7	Sedang

5.	Bk penyelsaian masalah	7	Sulit
6.	Algortima pemrograman	8	mudah
7.	Jejak bermedia di gital	8	Sedang
8.	Pemanfaatan pernakcta di gital	8	Sulit
9.	Uji kompotesi	9	Mudah
10.	Pengolahan data dasar	9	Sedang
11.	Mengenal data	9	Sulit

#### 4.1.7 Analisis Biaya

Untuk mendukung proses perancangan dan pengembangan aplikasi pembelajaran informatika berbasis web menggunakan algoritma *Quick Sort*, diperlukan sejumlah kebutuhan yang harus dipenuhi, baik dari segi perangkat keras, perangkat lunak, maupun sumber daya pendukung lainnya. Biaya ini mencakup seluruh komponen yang diperlukan hingga tahap pengujian aplikasi. Adapun detail biaya yang dibutuhkan dalam proyek ini adalah sebagai berikut:

Tabel 4.3 Analisis Biaya

No.	Jenis Kebutuhan	Biaya
1.	Laptop Acer	Rp. 6.000.000,
2.	Kuota Internet	Rp. 200.000,
3.	Buku	Rp. 200.000
4.	Analisi Kebutuhan	Rp. 500.000
5.	Programing	Rp. 300.000
6.	Desain	Rp. 100.000
Jumlah		Rp. 7.300.000

#### 4.2 Perancangan

Sebelum pembuatan aplikasi, perancangan sistem dilakukan untuk memastikan bahwa fitur dan alur kerja yang dikembangkan sesuai dengan

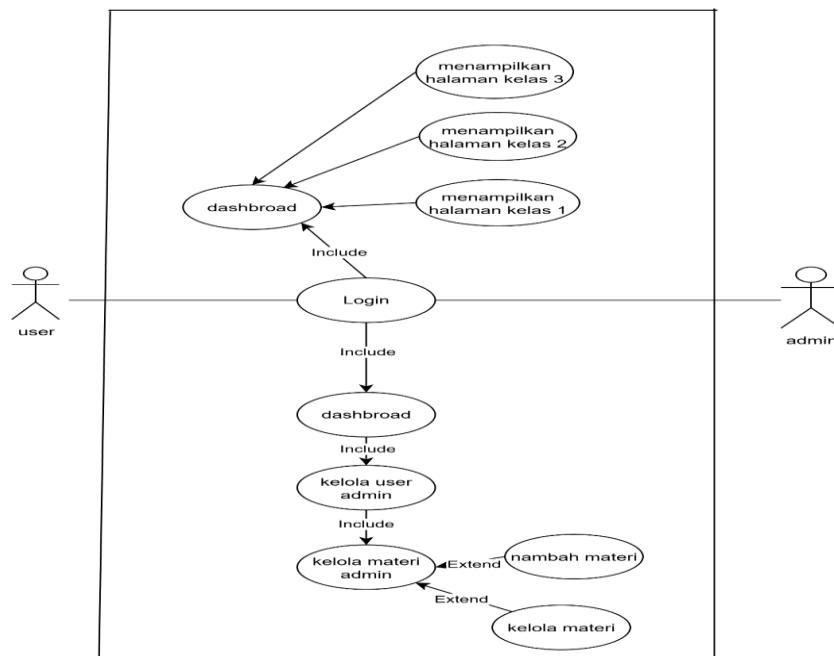
kebutuhan pengguna, yaitu siswa dan admin di SMP Plus Multazam. Proses ini melibatkan pembuatan model *UML* yang terdiri dari use case diagram untuk menggambarkan interaksi pengguna dengan sistem, activity diagram untuk menunjukkan alur aktivitas dalam aplikasi, serta class diagram yang merepresentasikan struktur data dan hubungan antar entitas.

#### 4.2.1 Perancangan sistem *UML*

*Unified Modeling Language (UML)* merupakan standar pemodelan yang digunakan untuk menggambarkan sistem perangkat lunak secara visual. UML membantu pengembang, analis sistem, dan stakeholder untuk memahami alur, fungsi, serta struktur sistem secara lebih terstruktur dan terorganisir. Dengan menggunakan *UML*, komunikasi antar pihak yang terlibat dalam pengembangan sistem menjadi lebih efektif karena setiap diagram memiliki tujuan dan fokus yang berbeda, seperti menggambarkan proses bisnis.

##### 1. Use Case Diagram

Pada rancangan *use case diagram* bertujuan untuk menggambarkan interaksi antara aktor dengan aplikasi pembelajaran informatika berbasis web.



Gambar 4. 1 Use Case Diagram User

Penejelasan use case diagram di atas dapat di lihat dari tabel berikut:

a. Deskripsi Aktor

Tabel 4.4 deskripsi aktor

Aktor	Deskripsi
Admin	Admin dalam aplikasi pembelajaran informatika berbasis web berperan sebagai pengelola utama sistem yang bertanggung jawab untuk mengatur dan memelihara seluruh data pengguna, termasuk guru dan siswa, serta memastikan semua fitur dalam aplikasi berjalan dengan baik. Admin memiliki akses penuh ke sistem, seperti menambahkan, mengedit, dan menghapus akun pengguna, serta melakukan pengawasan terhadap aktivitas dalam aplikasi.
User	User dalam aplikasi pembelajaran informatika berbasis web terdiri dari tiga jenis pengguna utama, yaitu siswa, guru, dan admin, yang masing-masing memiliki peran dan hak akses berbeda sesuai fungsinya dalam sistem. Siswa sebagai <i>user</i> utama menggunakan aplikasi untuk mengakses materi pembelajaran.

b. Deskripsi Use Case

Tabel 4.4 Use Case

Use case	Deskripsi
<i>login</i>	Proses awal yang di lakukan pengguna sebelum masuk ke dalam aplikasi, lalu sistem akan melakukan verifikasi data.
Dashboard	Tampilan utama aplikasi setelah <i>login</i> yang menampilkan informasi dasar mengenai aplikasi.
Kelas 1	Siswa atau guru dapat mengakses materi dan mempelajari materi, dan ditujukan untuk siswa tingkat awal dan berisi

	materi dasar informatika, seperti pengenalan komputer dan dasar logika algoritma.
Kelas 2	menyajikan materi lanjutan, seperti struktur data sederhana dan pengantar pemrograman, yang dirancang untuk memperluas wawasan dan keterampilan siswa setelah memahami dasar-dasar informatika pada kelas sebelumnya penyampaian materi dilakukan secara interaktif melalui materi digital, latihan soal, sehingga diharapkan dapat meningkatkan minat belajar siswa.
Kelas 3	Berisi materi pembelajaran tingkat lanjut yang difokuskan pada penguasaan konsep-konsep algoritma yang lebih kompleks, termasuk salah satunya adalah algoritma <i>Quick Sort</i> . Materi dalam kelas ini dirancang untuk memperkuat kemampuan berpikir logis dan komputasional siswa melalui topik-topik seperti pengurutan data, pencarian (searching), serta penerapan algoritma dalam pemrograman sederhana.
Logout	adalah fitur yang digunakan untuk menghentikan sesi login pengguna, sehingga pengguna tersebut tidak lagi memiliki akses ke fitur-fitur atau data pribadi dalam aplikasi hingga mereka login kembali.

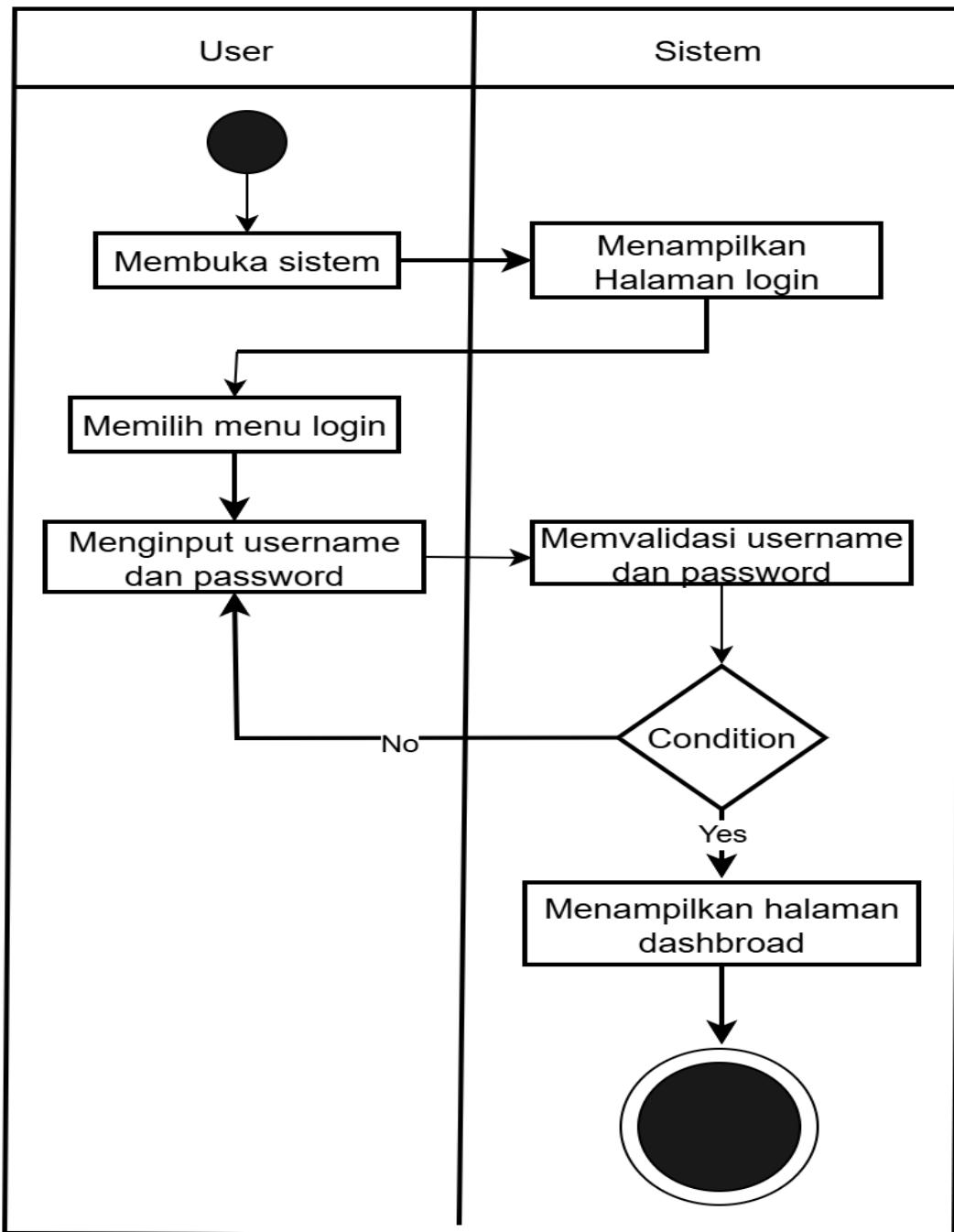
#### 4.2.2 Perancangan Activity Diagram

##### 2. *Activity Diagram*

Berikut adalah *activity diagram* pada aplikasi pembelajaran informatika sebagai berikut:

###### a. *Activity Diagram Login User*

*Activity diagram login* menggambarkan alur aktivitas atau proses yang terjadi dalam sistem, mulai dari awal hingga akhir, serta menunjukkan bagaimana satu aktivitas mengalir ke aktivitas berikutnya. Diagram ini menunjukkan langkah-langkah atau tahapan proses login.

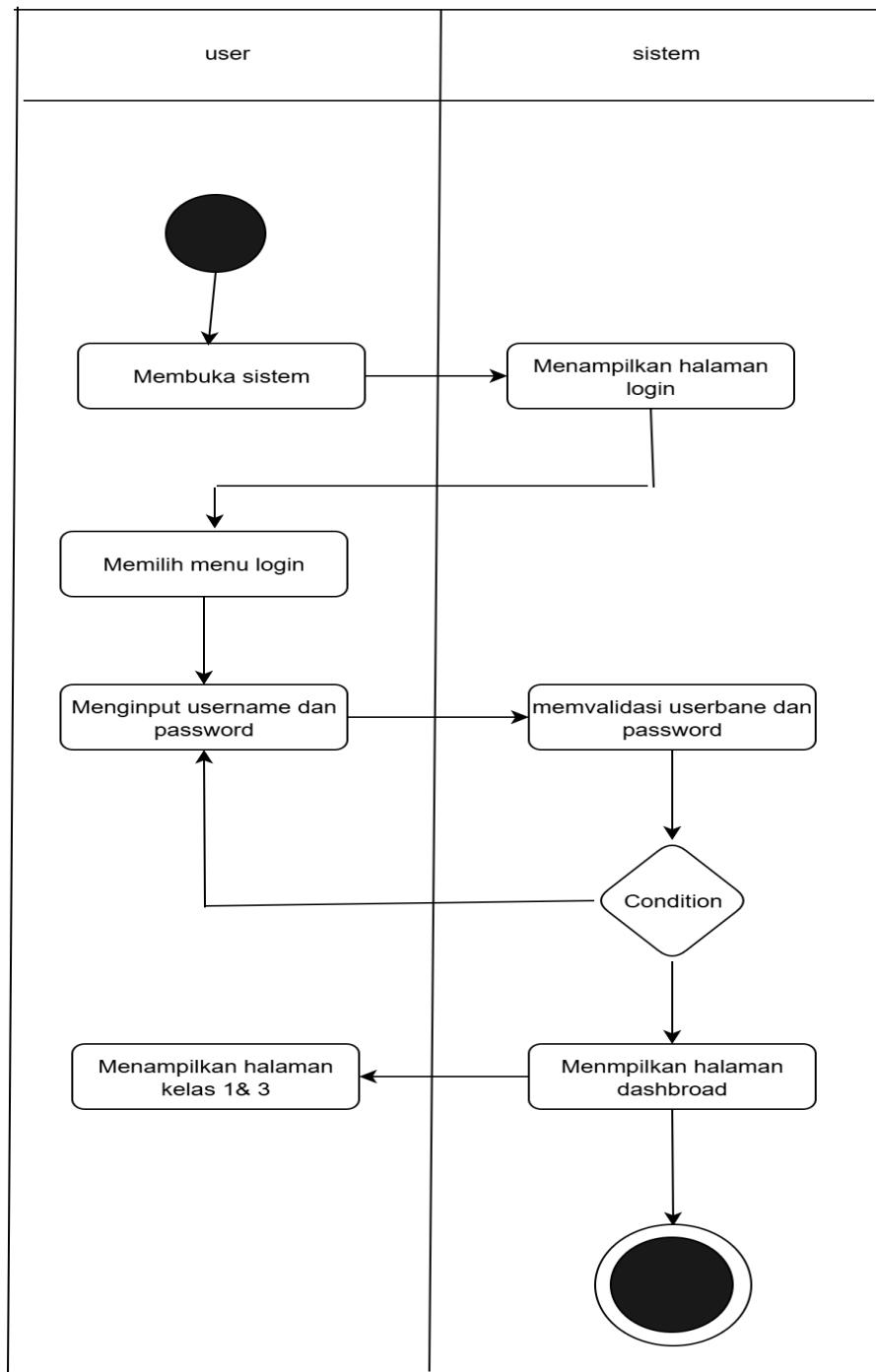


Gambar 4. 2 Activity Diagram Login

Pada gambar 4.2 menunjukan alur *user* yang akan melakukan *login* Langkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad.

b. *Activity Diagram Dashboard*

Diagram dashboard adalah alat bantu visual yang sangat penting dalam sistem informasi modern. Dengan menyajikan data dalam bentuk visual yang terorganisir dan mudah dimengerti, dashboard membantu pengguna.

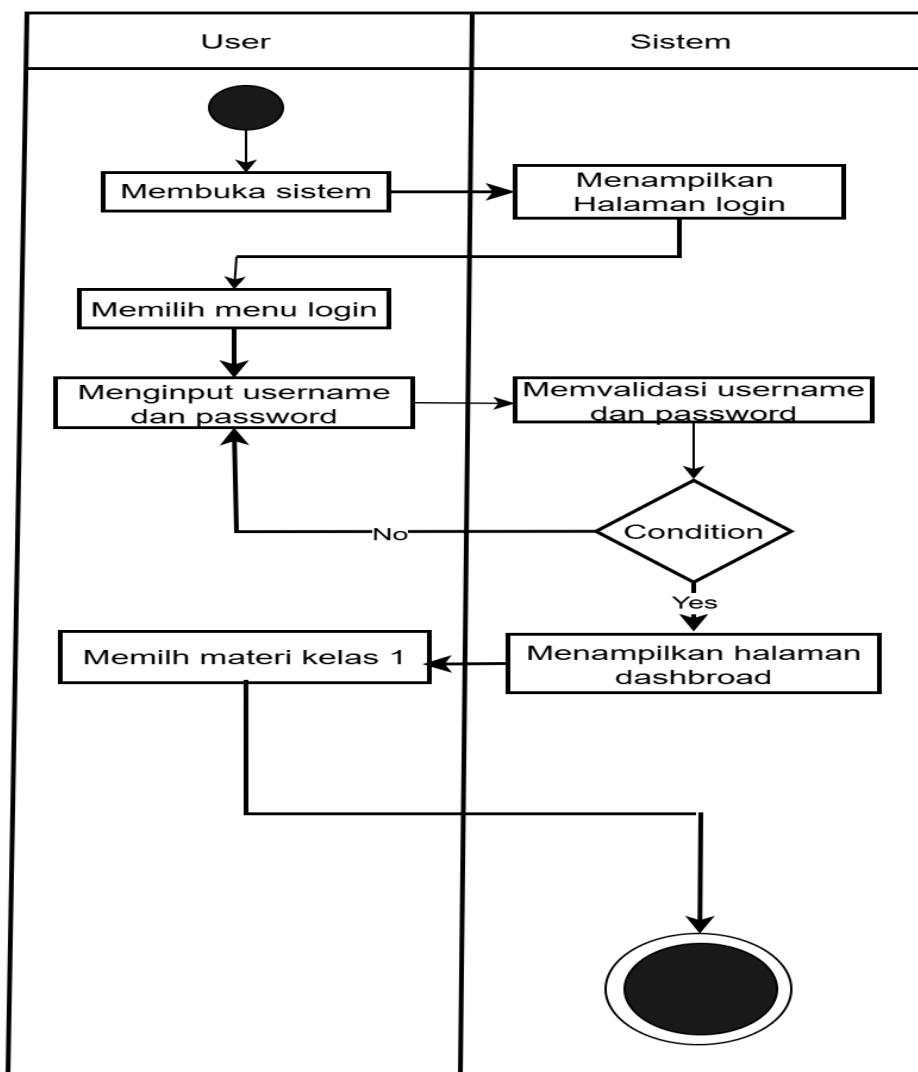


Gambar 4. 3 Activity Diagram Dashbroad

Pada gambar 4.3 menunjukan alur proses *user* melihat halaman dashbroad Langkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad dan *user* akan memilih menu dashbroad di mana dashbroad tersebut akan memiliki pilihan kelas 1- kelas 3 dan jika memilih salah satu kelas akan muncul materi yang sesuai di pilih.

c. *Activity Diagram* Kelas 1

*Activity Diagram* Kelas 1 adalah diagram yang menggambarkan alur siswa login hingga sistem menampilkan materi ajar kelas 1.

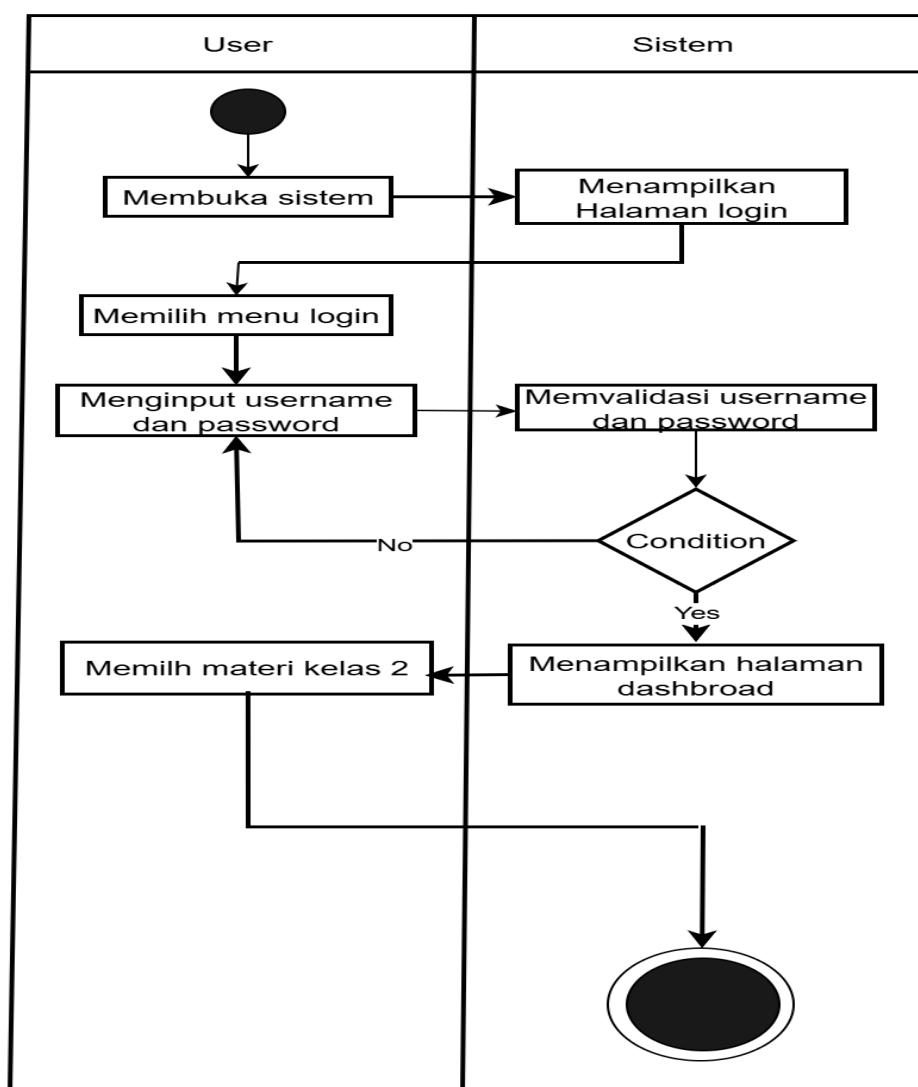


Gambar 4. 4 Activity Diagram Kelas 1

Pada gambar 4.4 menunjukkan alur proses *user* melihat halaman berandaLangkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad lalu *user* memilih kelas 1 dan sistem akan menampilkln materi ajar kelas 1.

d. *Activity Diagram Kelas 2*

Activity Diagram Kelas 2 adalah alat bantu visual yang digunakan untuk menunjukkan alur siswa dalam mengakses materi kelas 2 dengan langkah-langkah yang runtut dan jelas.

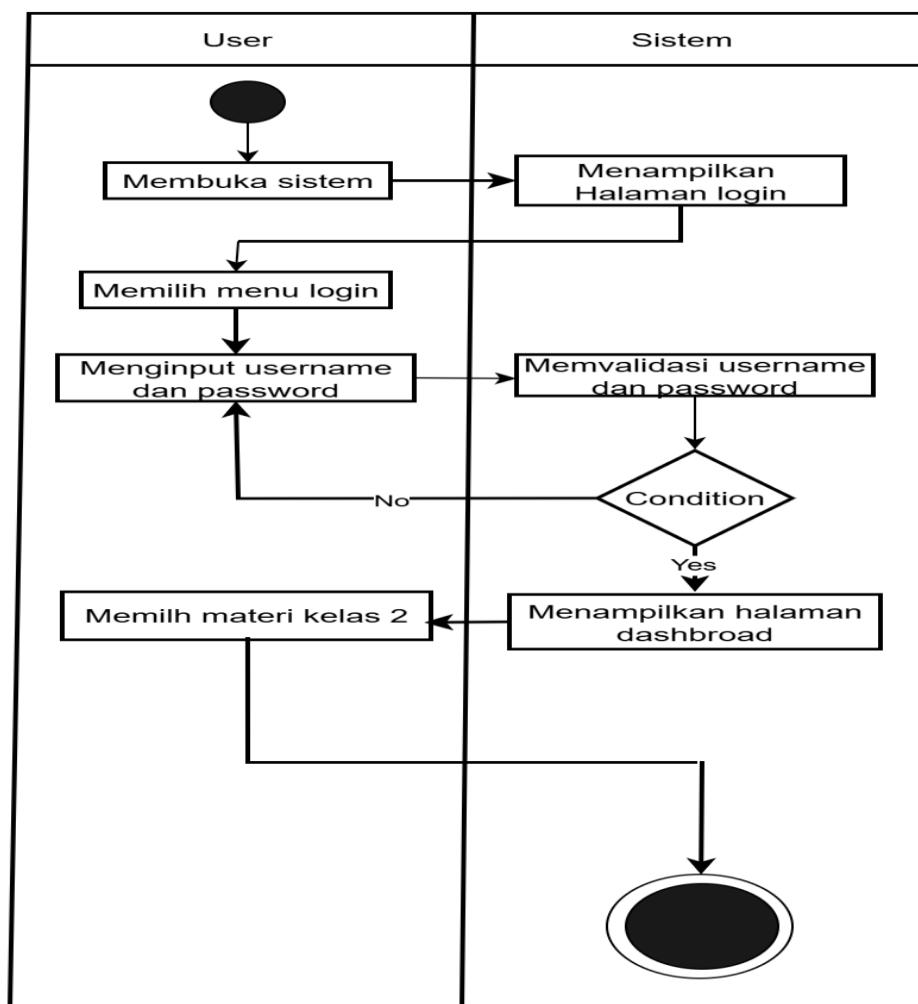


Gambar 4. 5Activity Diagram Kelas 2

Pada gambar 4.5 menunjukan alur proses *user* melihat halaman beranda Langkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad lalu *user* memilih kelas 1 dan sistem akan menampilkln materi ajar kelas 2.

e. *Activity diagram* Kelas 3

*Activity Diagram* Kelas 3 adalah alat bantu visual yang digunakan untuk memodelkan proses siswa ketika membuka sistem sampai sistem menampilkan materi ajar kelas 3.

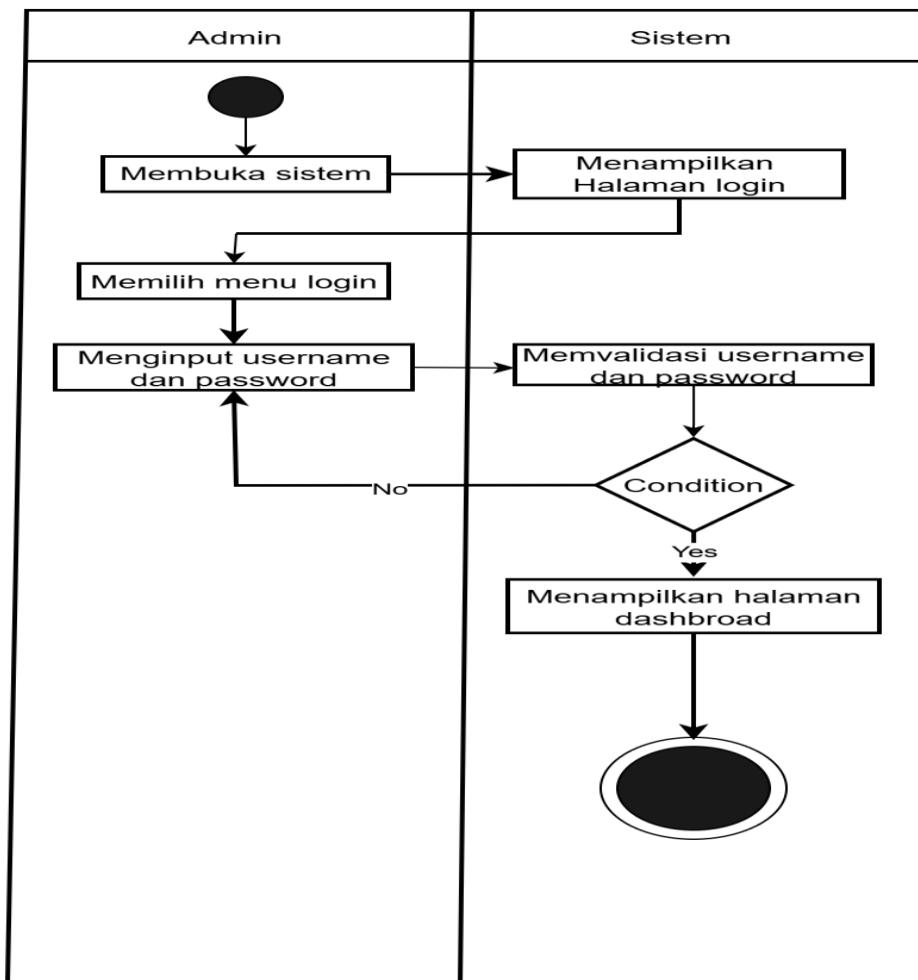


Gambar 4. 6Activity Diagram Kelas 3

Pada gambar 4.6 menunjukkan alur proses *user* melihat halaman berandaLangkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad lalu *user* memilih kelas 1 dan sistem akan menampilkln materi ajar kelas 3.

f. *Activity diagram login Admin*

*Activity Diagram Login Admin* adalah alat bantu visual yang digunakan untuk menggambarkan alur aktivitas admin saat masuk ke sistem, mulai dari memasukkan username dan password hingga sistem memvalidasi dan memberikan akses ke dashboard admin.

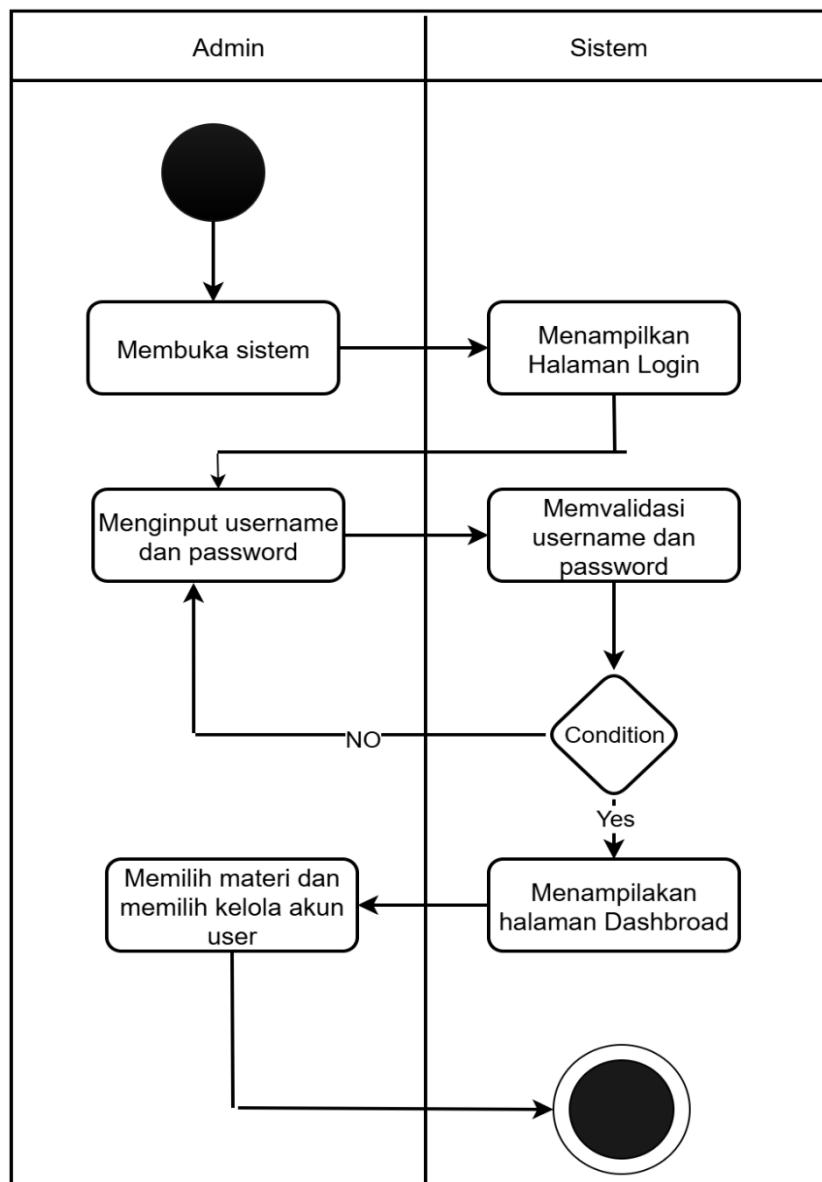


Gambar 4. 7Activity Diagram Login Admin

Pada gambar 4.7 menunjukkan alur *user* yang akan melakukan *login*. Langkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad.

g. *Activity diagram Dashbroad Admin*

Activity Diagram Dashboard Admin adalah alat bantu visual yang digunakan untuk memperlihatkan alur aktivitas admin setelah berhasil login.

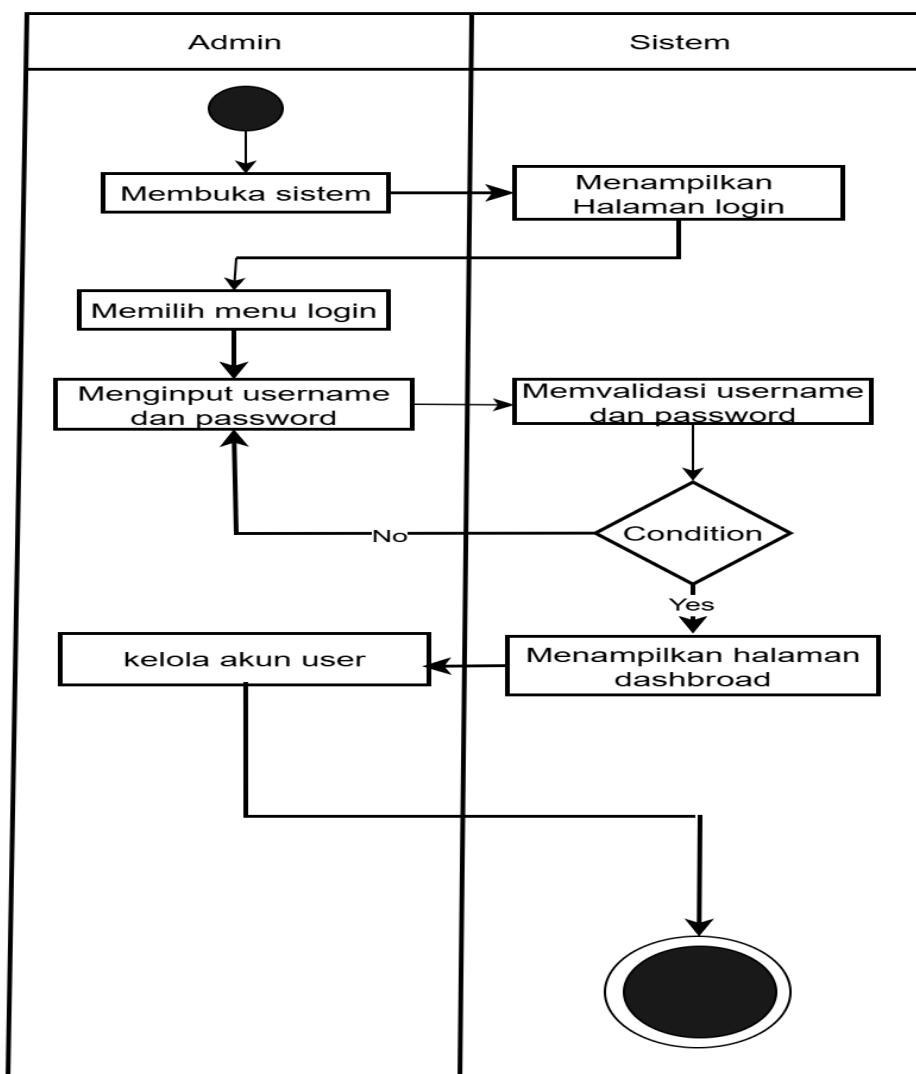


Gambar 4. 8 Activity Diagram Dashbroad Admin

Pada gambar 4.8 menunjukan alur proses admin melihat halaman beranda Langkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad admin di mana dashbroad admin memiliki mengklik kelola user.

#### h. *Activity Diagram Kelola Akun*

*Activity Diagram* Kelola User Admin adalah alat bantu visual yang digunakan untuk menggambarkan alur aktivitas admin dalam mengelola data pengguna, mulai dari menambah, mengedit, hingga menghapus akun sesuai kebutuhan sistem.

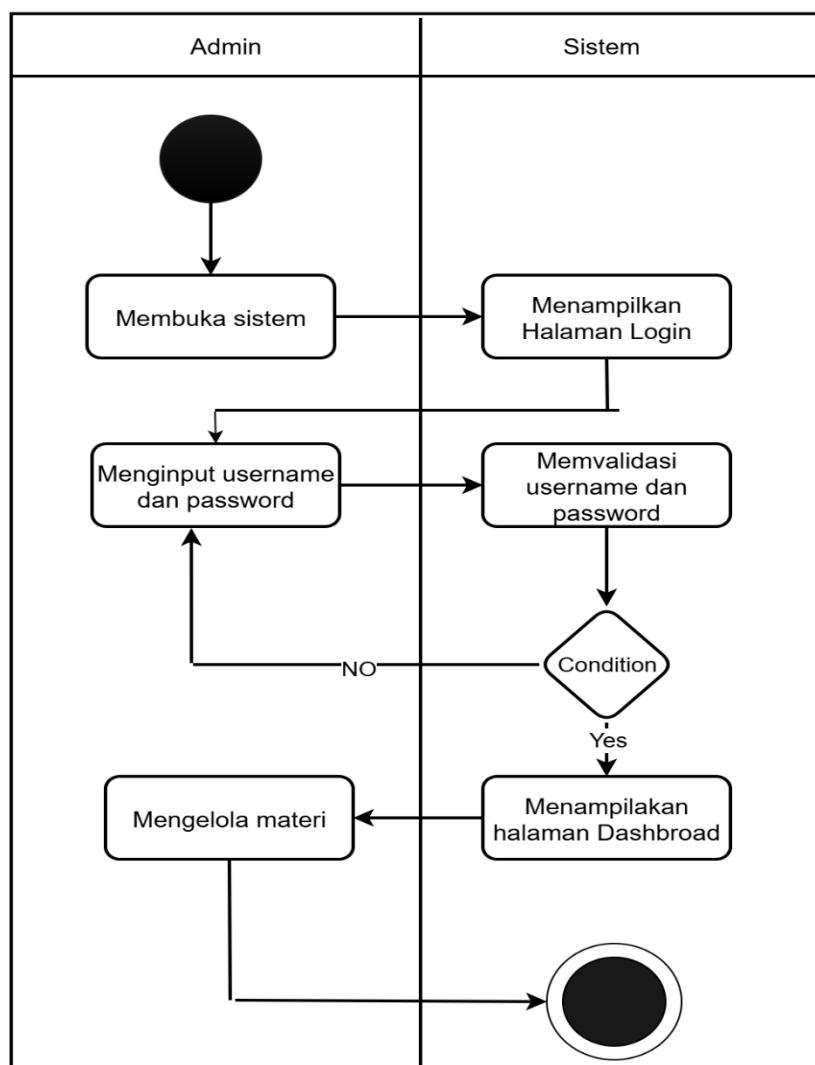


Gambar 4. 9Activity Diagram Kelola Akun

Pada gambar 4.9 menunjukan alur proses admin melihat halaman beranda Langkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad maka admin bisa melilih kelola akun user yang dimana admin bisa meng edit atau menghapus akun user.

i. *Activity Diagram Upload Materi Admin*

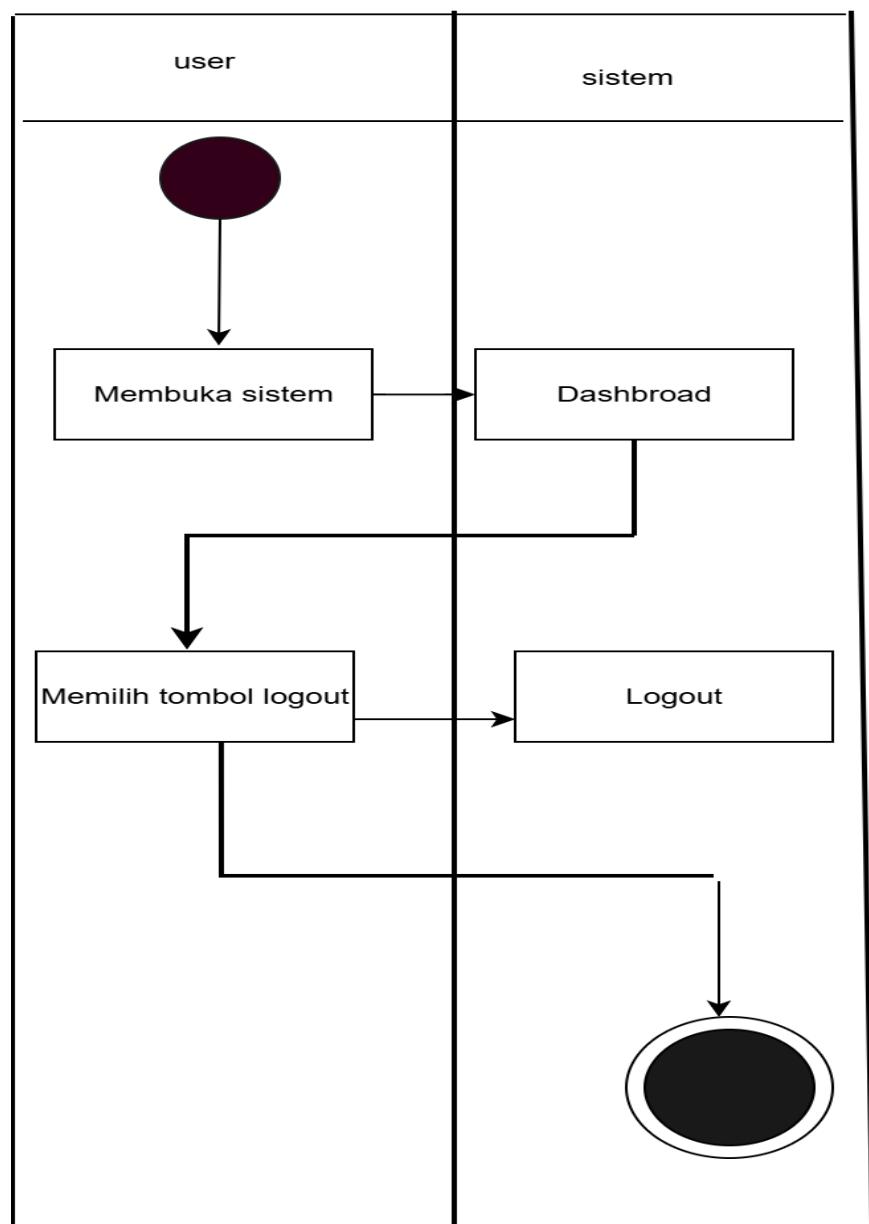
*Activity Diagram* Upload Materi Admin adalah alat bantu visual yang digunakan untuk menggambarkan alur aktivitas admin saat menambahkan materi pembelajaran baru ke dalam system.



Gambar 4. 10 Activity Diagram Upload Materi Admin

Pada gambar 4.11 menunjukan alur proses admin melihat halaman beranda Langkah pertama membuka sistem menampilkan halaman *login* lalu melilih menu *login* lalu *user* menginputkan *user name* dan *password* lalu sistem memvalidasi *user name* dan *password* jika tidak sesuai maka sistem akan bilang tidak, jika sesuai maka sistem akan menampilkan halaman dashbroad dan admin mengklik upload materi yang di mana admin bisa juga menghapus atau mengedit materi.

j. *Activity diagram logout*

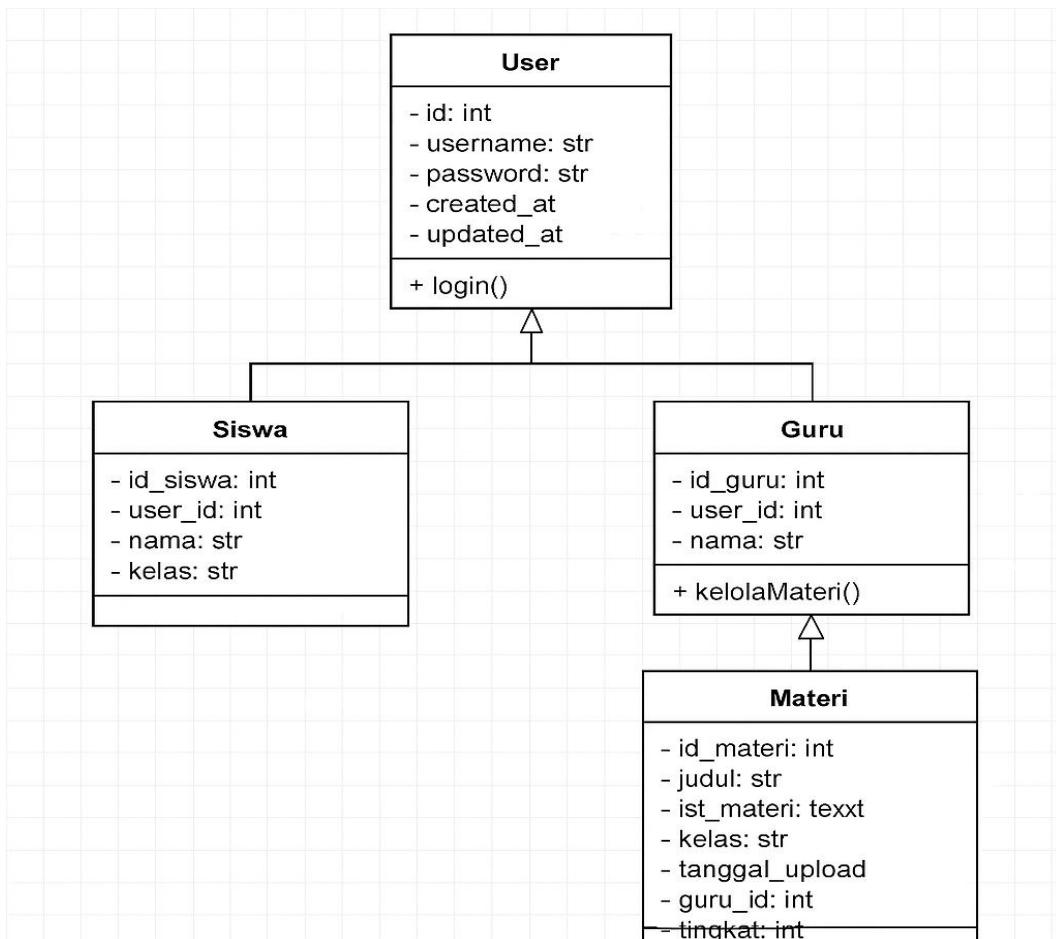


Gambar 4. 11 *Activity Diagram Logout*

Pada gambar 4.12 *user* akan melakukan proses *logout* pertama-tama mebuka sistem lalu memilih menu *logout* habis itu terus *logout*.

#### 4.2.3 Class daigaram

Class diagram adalah diagram dalam *UML* yang menggambarkan struktur statis sistem berupa kelas, atribut, dan metode. Diagram ini menunjukkan hubungan antar kelas seperti pewarisan (inheritance) dan asosiasi.



Gambar 4. 12Class daigram

#### 4.2.4 Struktut tabel

Struktur tabel adalah rancangan atau susunan data dalam basis data relasional yang digunakan untuk menyimpan, mengelola, dan menghubungkan informasi secara sistematis. Pada aplikasi pembelajaran informatika berbasis web ini, struktur tabel dirancang menggunakan MySQL dan terdiri dari kumpulan tabel yang

merepresentasikan entitas utama seperti pengguna (*users*), materi pembelajaran (*materials*), dan data pendukung lainnya. Setiap tabel terdiri dari kolom (*field*) yang menyimpan atribut tertentu, misalnya nama pengguna, judul materi, tingkat kesulitan, atau tanggal pembaruan, sedangkan setiap baris (*record*) menyimpan nilai aktual dari atribut tersebut. Hubungan antar tabel diatur melalui kunci primer (*primary key*) untuk identifikasi unik setiap record dan asing kunci (*foreign key*) untuk menghubungkan data antar tabel. Perancangan struktur tabel yang baik memungkinkan penerapan algoritma *Quick Sort* untuk mengurutkan data, seperti menyusun materi pembelajaran berdasarkan tingkat kesulitan atau tanggal pembaruan.

### 1. Perancangan tabel *user*

Tabel 4.1 tabel user

Fiel	Tipe Data	Keterangan
Id	INT (PK, AI)	ID <i>user</i>
<i>user name</i>	VARCHAR (50)	Nama pengguna
password	VARCHAR (255)	Password yang telah di-hash
created_at	DATETIME	Tanggal pembuatan akun
updated_at	DATETIME	Tanggal update akun

### 2. Perancangan tabel siswa

Tabel 4.1 tabel siswa

Fiel	Tipe Data	Keterangan
id_siswa	INT (PK, AI)	ID siswa
<i>user_id</i>	INT (FK)	Relasi ke tabel <i>user</i> s
nama_lengkap	VARCHAR (100)	Nama lengkap siswa
kelas	VARCHAR (10)	Kelas (misal: 1, 2, atau 3)

### 3. Perancangan tabel guru

Tabel 4.1 tabel guru

Fiel	Tipe data	Keterangan
id_guru	INT (PK, AI)	ID guru
user_id	INT (FK)	Relasi ke tabel user
nama_guru	VARCHAR (100)	Nama lengkap guru

### 4. Perancangan tabel materi

Tabel 4.1 tabel materi

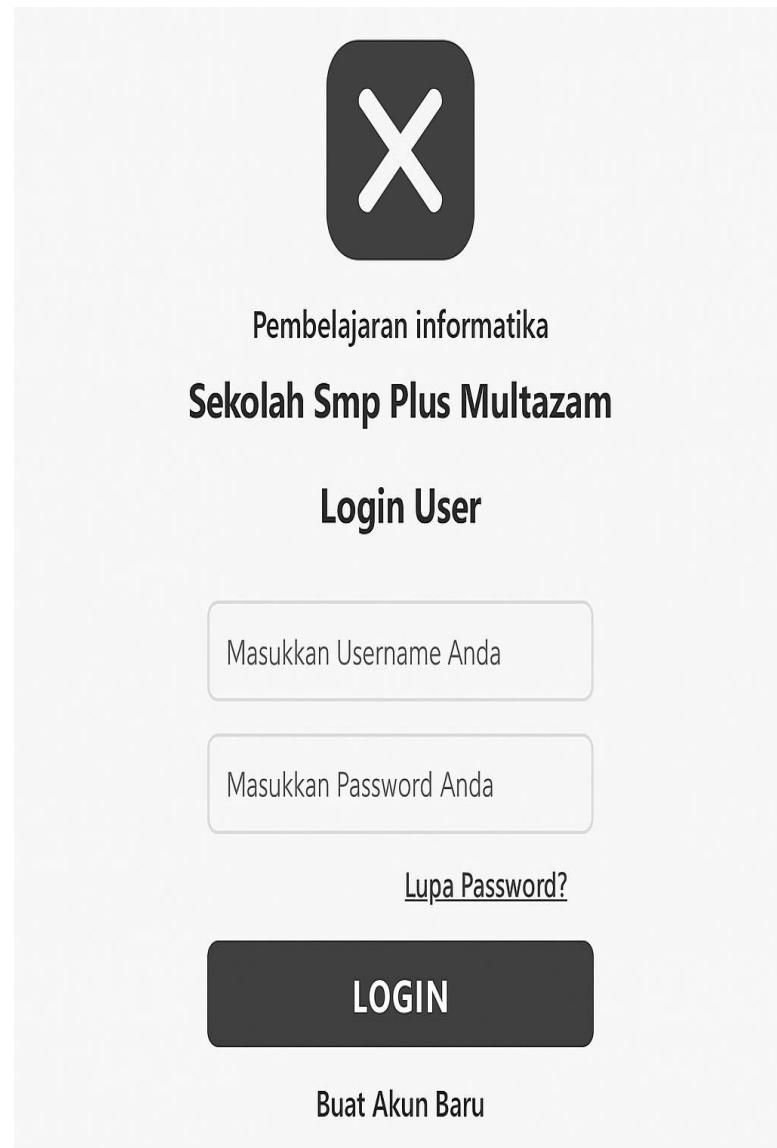
Fiel	Tipe Data	Keterangan
id_materi	INT (PK, AI)	ID materi
Judul	VARCHAR (150)	Judul materi
isi_materi	TEXT	Konten materi
kelas	VARCHAR (10)	Kelas target (1/2/3)
tanggal_upload	DATE	Tanggal unggah
guru_id	INT (FK)	Relasi ke guru
tingkat	INT	Tingkat kesulitan (untuk sorting)

#### 4.2.5 Perancangan Desain Antarmuka

Desain antarmuka merupakan proses perancangan tampilan visual dari suatu sistem atau aplikasi yang berfungsi sebagai penghubung antara pengguna dan sistem, untuk memudahkan dalam menjalankan fungsi-fungsi yang disediakan oleh aplikasi. Antarmuka yang baik tidak hanya memperhatikan aspek estetika, tetapi juga kenyamanan, konsistensi, dan kemudahan penggunaan bagi pengguna. Tujuan utama dari desain antarmuka adalah menciptakan pengalaman pengguna yang intuitif dan efisien saat berinteraksi dengan aplikasi. Berikut adalah desain antarmuka yang dibuat untuk mendukung kebutuhan tersebut.

1. Desain tampilan antarmuka login user

Berikut ini adalah tampilan antarmuka halaman login admin pada aplikasi penyesuaian format skripsi yang akan dibuat:

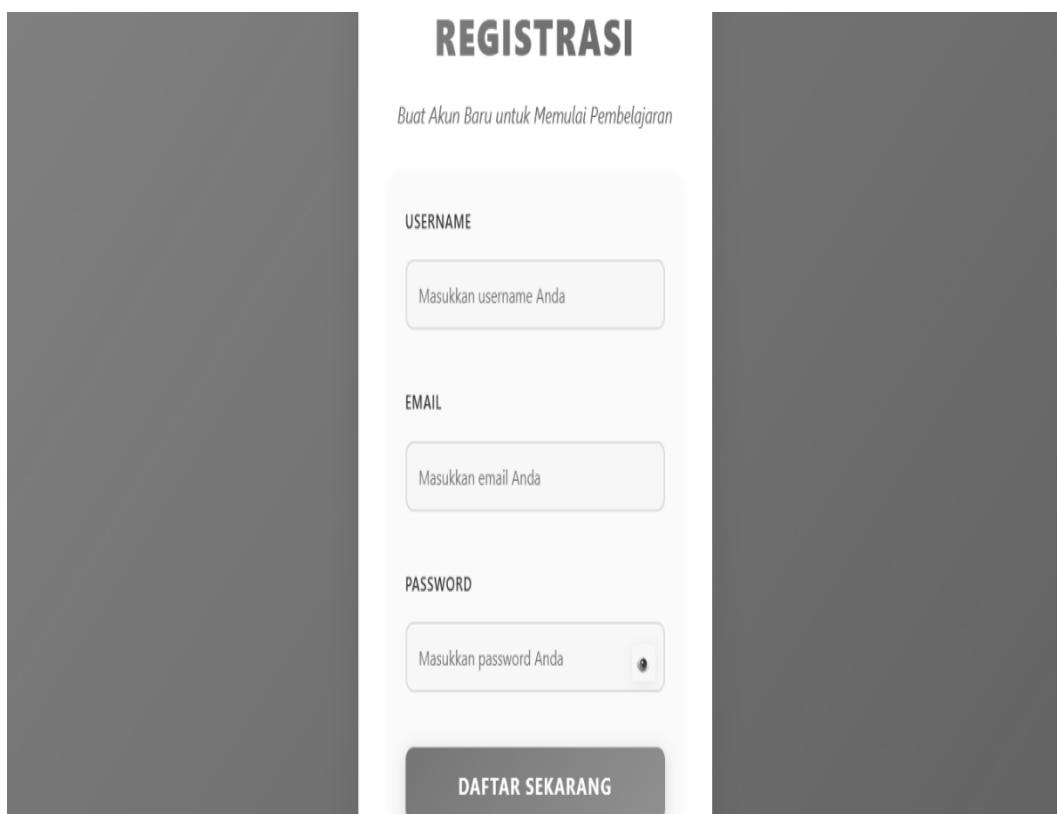


Gambar 4. 13 Desain Antarmuka User

Antarmuka login admin ini terdiri dari dua input field utama, yaitu "username" dan "password". Pengguna diminta untuk memasukkan informasi akun dengan benar sebelum menekan tombol "sign in" yang berada di bagian bawah. Selain itu, tersedia tautan "forgot password?" sebagai opsi bagi pengguna yang lupa kata sandi.

2. Desain tampilan antarmuka halaman register

Berikut ini adalah tampilan antarmuka halaman register pada aplikasi pembelejaran informatika.

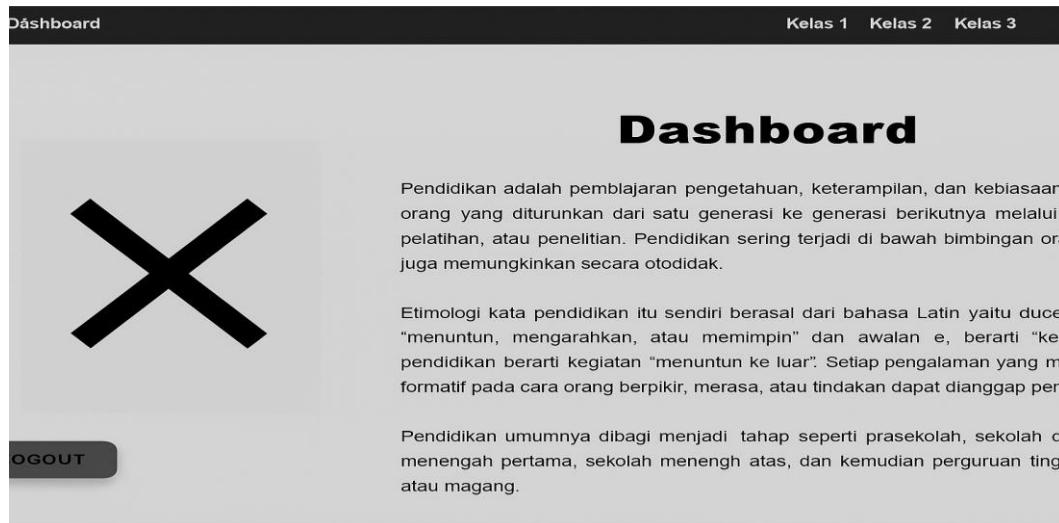


Gambar 4. 14 Buat Akun Baru

Antarmuka registrasi memudahkan pengguna dalam membuat akun baru pada aplikasi pembelajaran. Halaman ini menampilkan formulir terstruktur dengan tiga isian utama, yaitu *username*, *email*, dan *password*. Setiap kolom dilengkapi petunjuk teks di dalam kotak isian untuk membantu pengguna mengisi data dengan benar. Pada kolom password terdapat ikon mata yang memungkinkan pengguna menampilkan atau menyembunyikan karakter sandi. Di bagian bawah tersedia tombol abu-abu “DAFTAR SEKARANG” yang berfungsi untuk mengirim data pendaftaran ke sistem.

3. Desain tampilan antarmuka dashboard

Berikut ini adalah tampilan antarmuka dashboard pada aplikasi pembelajaran informatika

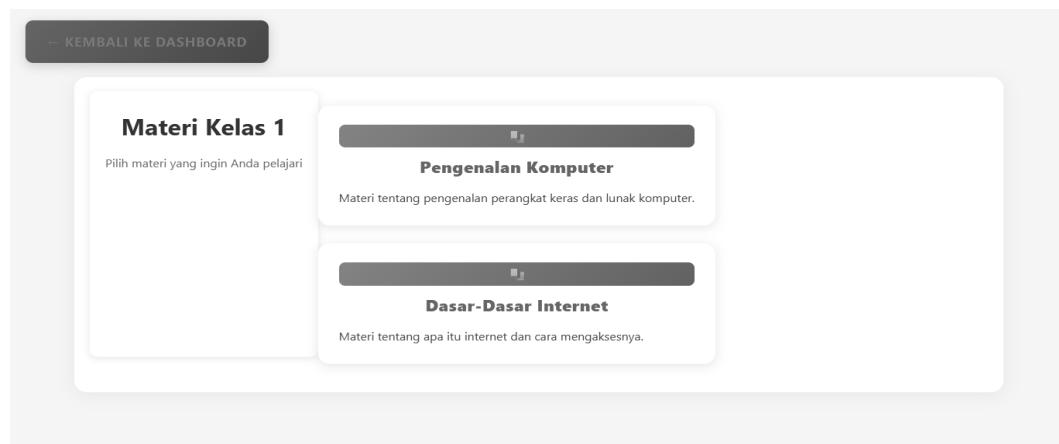


Gambar 4. 15 Desain Antarmuka Dashboard

Antarmuka dashboard pada aplikasi pembelajaran menampilkan informasi utama mengenai pendidikan secara umum. Terdapat judul besar “Dashboard” di bagian atas konten, dengan tiga paragraf penjelasan di sisi kanan layar yang membahas pengertian, etimologi, dan tahapan pendidikan. Di bagian kiri, Navigasi utama terletak di bagian atas serta menu kelas berupa “Kelas 1”, “Kelas 2”, dan “Kelas 3” di sisi kanan.

#### 4. Desain tampilan antarmuka materi kelas 1

Berikut ini adalah tampilan antarmuka materi kelas 1 pada halaman pembelajaran informatika.

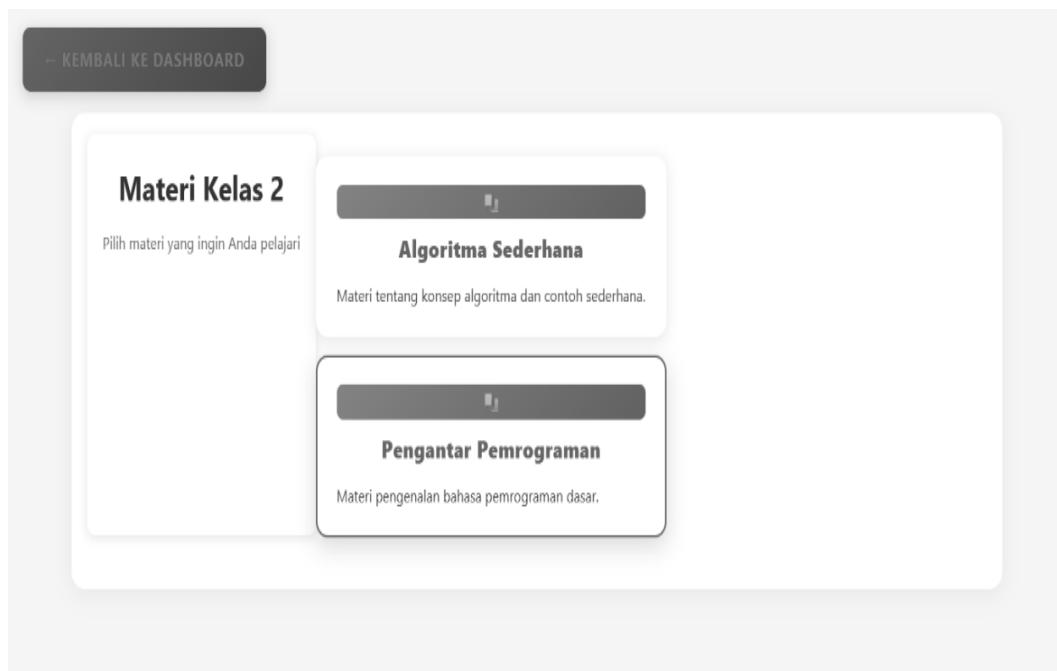


Gambar 4. 16 Desain Antarmuka Materi Kelas 1

Antarmuka halaman pembelajaran kelas 1 pengguna akan melihat halaman yang dirancang khusus untuk siswa kelas 1. Di sisi kiri halaman, terdapat panel informasi dengan judul "Materi Kelas 1", disertai teks pendukung yang mengarahkan pengguna untuk memilih materi yang ingin dipelajari. Di bagian kanan, terdapat dua kartu materi utama yang disusun secara vertikal, masing-masing mewakili topik pembelajaran dasar di bidang teknologi informasi: Pengenalan KomputerDasar-Dasar Internet.

##### 5. Desain tampilan antarmuka materi kelas 2

Berikut ini adalah tampilan antarmuka materi kelas 2 pada halaman pembelajaran informatika.

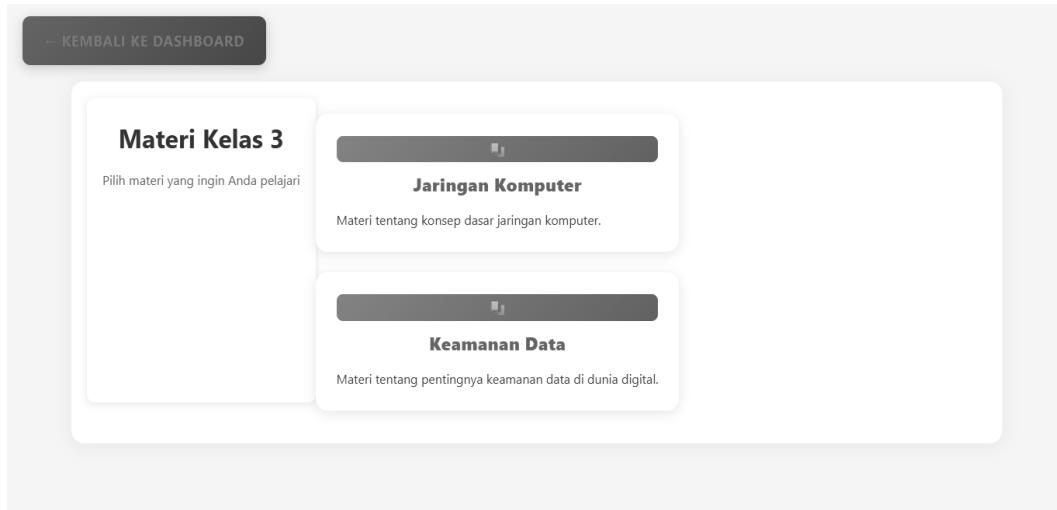


Gambar 4. 17 Desain Antarmuka Materi Kelas 2

Antarmuka halaman materi kelas 2, pengguna akan diarahkan ke tampilan beranda yang memperlihatkan daftar materi belajar untuk kelas tersebut. Di bagian kiri layar, terdapat panel informasi berjudul "Materi Kelas 2" yang memberikan arahan singkat bahwa pengguna dapat memilih materi yang ingin dipelajari. Di sebelah kanan panel tersebut, terdapat dua kartu materi utama yang disesuaikan dengan kebutuhan siswa kelas 2, yaitu: Algoritma Sederhana, Pengantar Pemrograman, yang memperkenalkan bahasa pemrograman dasar.

6. Desain tampilan antarmuka materi kelas 3

Berikut adalah tampilan antarmuka materi kelas 3 pada halaman pembelajaran informatika

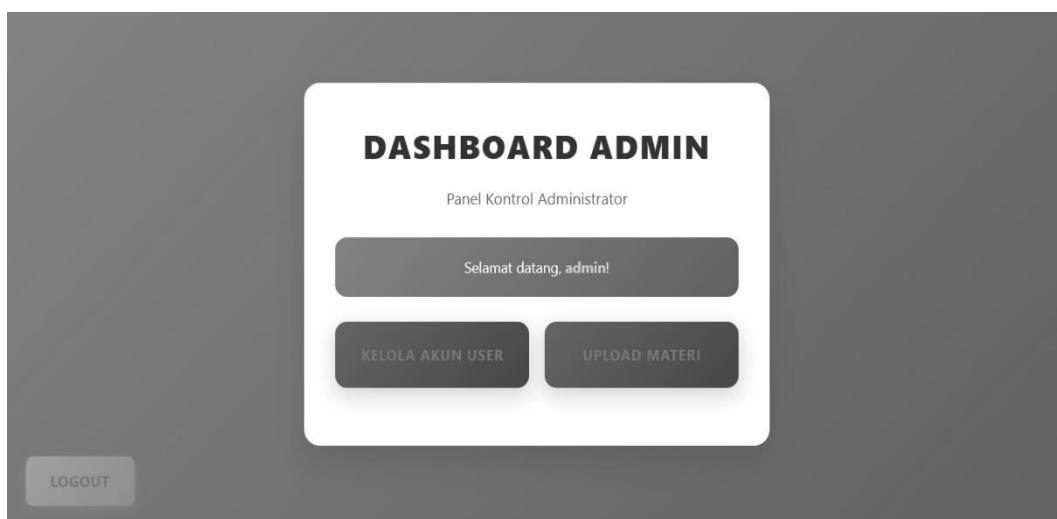


Gambar 4. 18 Desain Antarmuka Materi Kelas 3

antarmuka halaman materi kelas 3, pengguna akan melihat halaman yang dirancang khusus untuk siswa kelas 3. Di sisi kiri halaman, terdapat panel informasi dengan judul "Materi Kelas 3", disertai teks pendukung yang mengarahkan pengguna untuk memilih materi yang ingin dipelajari.

7. Desain antarmuka dashbroad admin

Berikut adalah tampilan dashbroad admin pada halaman pembelajaran informatika yang akan di buat:

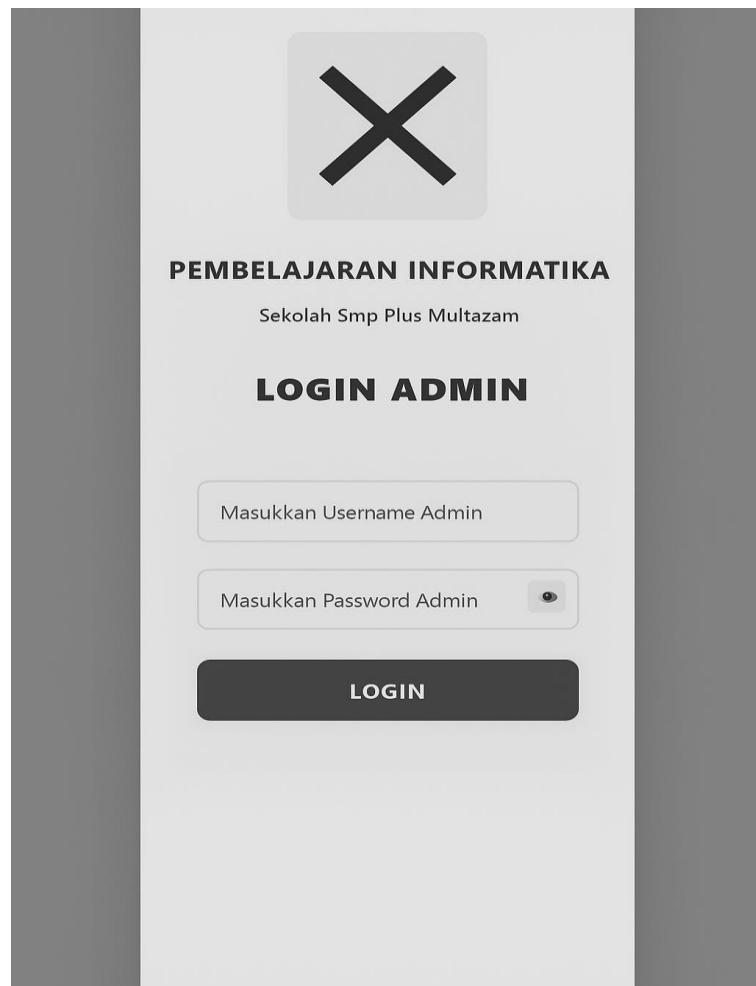


Gambar 4. 19 Desain Antarmuka Dashbroad Admin

Antarmuka dashboard admin pada aplikasi pembelajaran menampilkan panel utama yang ditujukan bagi pengguna dengan peran administrator. Di bagian tengah terdapat judul besar “DASHBOARD ADMIN” sebagai identitas halaman, dengan subjudul “Panel Kontrol Administrator” tepat di bawahnya. Tampilan ini juga menyajikan pesan sambutan “Selamat datang, admin!” Di bawahnya terdapat dua tombol fungsi utama yaitu “KELOLA AKUN USER” untuk mengelola data pengguna dan “UPLOAD MATERI” untuk menambahkan konten pembelajaran.

#### 8. Desain tampilan antarmuka *login admin*

Berikut adalah tampilan antarmuka login adamin pada halaman pembelajaran informatika yang akan di buat:



Gambar 4. 20 Desain Antarmuka Login Admin

Antarmuka login admin ini terdiri dari dua input field utama, yaitu "username" dan "password". Pengguna diminta untuk memasukkan informasi akun dengan benar sebelum menekan tombol "sign in" yang berada di bagian bawah. Selain itu, tersedia tautan "forgot password?" sebagai opsi bagi pengguna yang lupa kata sandi. Desain antarmuka ini bersifat minimalis, responsif, dan fokus pada fungsionalitas, sehingga memudahkan admin untuk masuk ke dalam sistem secara cepat dan aman.

#### 9. Desain tampilan antarmuka kelola user

Berikut tampilan antarmuka kelola user pada halaman bagian admin yang akan dibuat:

The screenshot shows a user management interface titled "KELOLA AKUN USER". On the left, there is a sidebar with the text "Panel Manajemen Pengguna Sistem" and a button labeled "← Kembali ke Dashboard Admin". The main area displays a table with the following data:

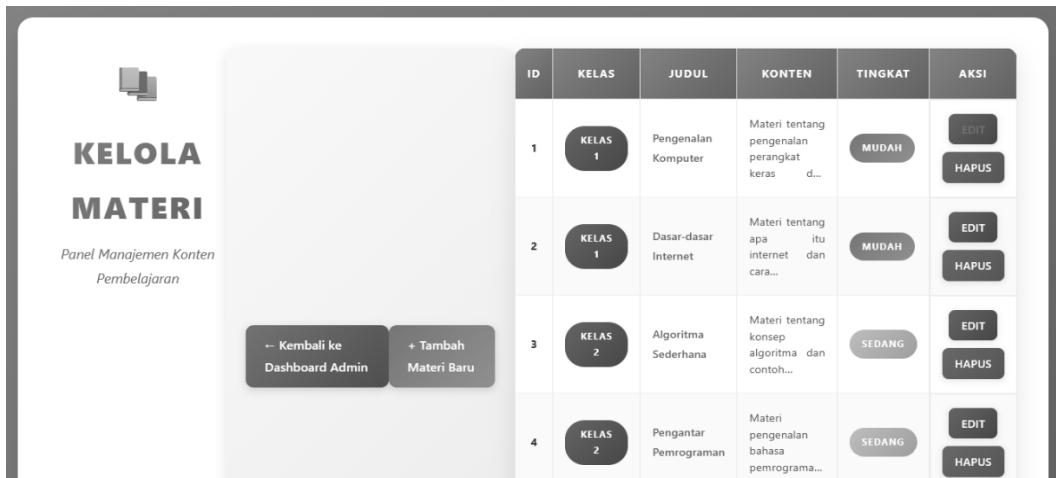
ID	USERNAME	EMAIL	ROLE	AKSI
1	admin	admin@email.com	ADMIN	Hapus
2	user	user@email.com	USER	Hapus
3	karim	karimhasbi40@gmail.com	USER	Hapus

Gambar 4. 21 Antarmuka Kelola User

Antarmuka kelola akun user menampilkan panel manajemen pengguna sistem yang memungkinkan admin untuk melihat dan mengatur akun pengguna. Di sisi kiri, terdapat judul besar “KELOLA AKUN USER” dengan subjudul “Panel Manajemen Pengguna Sistem” sebagai identifikasi halaman. Sementara itu, di sisi kanan ditampilkan tabel data user dalam format terstruktur, mencakup kolom ID, Username, Email, Role, dan Aksi. Setiap baris memuat informasi pengguna dan

dilengkapi dengan tombol aksi di sebelah kanan, yang berfungsi untuk melakukan pengeditan atau penghapusan data.

#### 10. desain antarmuka kelola materi



The screenshot shows a user interface titled "KELOLA MATERI" (Manage Materials). On the left, there's a sidebar with "Panel Manajemen Konten Pembelajaran". Below it are two buttons: "← Kembali ke Dashboard Admin" and "+ Tambah Materi Baru". The main area displays a table with four rows of material data:

ID	KELAS	JUDUL	KONTEN	TINGKAT	AKSI
1	KELAS 1	Pengenalan Komputer	Materi tentang pengenalan perangkat keras d... ...	MUDAH	<a href="#">EDIT</a> <a href="#">HAPUS</a>
2	KELAS 1	Dasar-dasar Internet	Materi tentang apa itu internet dan cara...	MUDAH	<a href="#">EDIT</a> <a href="#">HAPUS</a>
3	KELAS 2	Algoritma Sederhana	Materi tentang konsep algoritma dan contoh...	SEDANG	<a href="#">EDIT</a> <a href="#">HAPUS</a>
4	KELAS 2	Pengantar Pemrograman	Materi pengenalan bahasa pemprogramma...	SEDANG	<a href="#">EDIT</a> <a href="#">HAPUS</a>

Gambar 4. 22Desain Antarmuka Kelola Materi

Antarmuka kelola materi berfungsi sebagai panel manajemen konten pembelajaran yang memungkinkan admin untuk menambahkan, mengedit, atau menghapus materi ajar. Di bagian kiri layar, terdapat judul besar “KELOLA MATERI” dengan subjudul “Panel Manajemen Konten Pembelajaran”. Sisi kanan layar menampilkan tabel data materi dalam format yang terstruktur dan rapi. Tabel ini memuat beberapa kolom, antara lain ID, Kelas, Judul, Konten, Tingkat, dan Aksi. Kolom “Tingkat” mengindikasikan level kesulitan materi seperti “MUDAH” atau “SEDANG”. Pada kolom “Aksi”, terdapat dua tombol fungsional yaitu “EDIT” dan “HAPUS” untuk memodifikasi atau menghapus data materi.

## **BAB V**

### **IMPLEMENTASI DAN PENGUJIAN**

#### **5.1 Implementasi**

Setelah melaksanakan analisis dan perancangan maka selanjutnya adalah pengimplementasian untuk menjalankan analisis dan perancangan yang sudah dibuat ke dalam bentuk aplikasi.

##### **5.1.1 Listing program**

1. Listing program halaman login user

```
<?php
session_start();
if (isset($_SESSION['user_id'])) {
    header('Location: dashboard.php');
    exit();
}
$error = $_SESSION['login_error'] ?? '';
unset($_SESSION['login_error']);
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>
body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
    margin: 0;
}
```

2. Listing program halaman register akun

```
<?php
require_once 'includes/db.php';

$username = $_POST['username'] ?? "";
$email = $_POST['email'] ?? "";
$password = $_POST['password'] ?? "";

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Validasi sederhana
    if (!$username || !$email || !$password) {
        header('Location: register.php');
        exit();
    }
    // Cek username/email unik
    $stmt = $conn->prepare("SELECT id FROM users WHERE username = ? OR
email = ?");
    $stmt->bind_param('ss', $username, $email);
    $stmt->execute();
    $stmt->store_result();
    if ($stmt->num_rows > 0) {
        header('Location: register.php');
        exit();
    }
    // Hash password
    $hashed = password_hash($password, PASSWORD_DEFAULT);
    $stmt = $conn->prepare("INSERT INTO users (username, email, password)
VALUES (?, ?, ?)");
    $stmt->bind_param('sss', $username, $email, $hashed);
    $stmt->execute();
    header('Location: login.php');
```

```

    exit();
}
?>

```

### 3. Listing program dashboard user

```

</head>
<body>

<div class="header">
    <div class="nav">
        <a href="#" class="nav-dashboard">Dashboard</a>
        <div class="nav-right">
            <a href="kelas1.php">Kelas 1</a>
            <a href="kelas2.php">Kelas 2</a>
            <a href="kelas3.php">Kelas 3</a>
        </div>
    </div>
</div>

<div class="content" style="background:#e6f7fa; display:flex; align-items:center; justify-content:center; min-height:400px; padding:40px 0;">
    <div style="flex:0 0 380px; display:flex; justify-content:center; align-items:center;">
        
    </div>
    <div style="flex:1; max-width:700px; margin-left:40px;">
        <h1 style="text-align:center; font-size:2.8rem; font-family:'Montserrat',Arial,sans-serif; font-weight:900; margin-bottom:28px; letter-spacing:1px;">Dashboard</h1>
        <p style="font-size:1.1rem; line-height:1.7; text-align:justify; margin-top:10px;">

```

#### 4. Listing program materi Kelas 1

```
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

include 'includes/db.php';

// Ambil materi untuk kelas 1
$query = "SELECT * FROM materi WHERE kelas = 1 ORDER BY id";
$result = mysqli_query($conn, $query);
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Kelas 1 - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>
```

#### 5. Listing program materi Kelas 2

```
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}
```

```

include 'includes/db.php';

// Ambil materi untuk kelas 2
$query = "SELECT * FROM materi WHERE kelas = 2 ORDER BY id";
$result = mysqli_query($conn, $query);
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Kelas 2 - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>

```

## 6. Listing program materi Kelas 3

```

<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

include 'includes/db.php';

// Ambil materi untuk kelas 3
$query = "SELECT * FROM materi WHERE kelas = 3 ORDER BY id";
$result = mysqli_query($conn, $query);
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Kelas 3 - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">

```

```
<style>
```

## 7. Listong prorgram Nambah Materi

```
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

include 'includes/db.php';

// Cek apakah ada parameter id
if (!isset($_GET['id']) || empty($_GET['id'])) {
    header('Location: dashboard.php');
    exit();
}

$id = mysqli_real_escape_string($conn, $_GET['id']);
$query = "SELECT * FROM materi WHERE id = '$id'";
$result = mysqli_query($conn, $query);

// Cek apakah query berhasil dan ada data
if (!$result || mysqli_num_rows($result) == 0) {
    echo "<script>alert('Materi tidak ditemukan!');  
window.location.href='dashboard.php';</script>";
    exit();
}
```

## 8. Listing program Admin login

```
<?php
session_start();
require_once 'includes/db.php';

$username = $_POST['username'] ?? "";
$password = $_POST['password'] ?? "";

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $stmt = $conn->prepare("SELECT id, username, password FROM users
WHERE username = ? AND role = 'admin'");
```

```

$stmt->bind_param('s', $username);
$stmt->execute();
$result = $stmt->get_result();
if ($row = $result->fetch_assoc()) {
    if (password_verify($password, $row['password'])) {
        $_SESSION['admin_id'] = $row['id'];
        $_SESSION['admin_username'] = $row['username'];
        header('Location: admin_dashboard.php');
        exit();
    }
}
$error = 'Username atau password admin salah!';
$_SESSION['admin_login_error'] = $error;
header('Location: admin_login.php');
exit();
}
?>

```

## 9. Listing program Dashboard admin

```

<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Dashboard Admin - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>

```

```
body {
```

## 10. Listing program Admin materi

```
<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Upload Materi - Admin</title>
    <link rel="stylesheet" href="assets/style.css">
</head>
<body>
    <h2>Upload Materi</h2>
    <form action="admin_materi_add_process.php" method="post">
        <label>Kelas:</label><br>
```

## 11. Listing program Admin materi delet

```
?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';
$id = $_GET['id'] ?? '';
if ($id) {
    $stmt = $conn->prepare('DELETE FROM materi WHERE id=?');
    $stmt->bind_param('i', $id);
    $stmt->execute();
```

```

}

header('Location: admin_materi.php');

exit();

?>

```

## 12. Lsiting program Admin materi edit

```

<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';
$id = $_GET['id'] ?? '';
if ($_SERVER['REQUEST_METHOD'] === 'POST' && $id) {
    $kelas = $_POST['kelas'] ?? '';
    $judul = $_POST['judul'] ?? '';
    $konten = $_POST['konten'] ?? '';
    $tingkat = $_POST['tingkat'] ?? 'mudah';
    if ($kelas && $judul && $konten && $tingkat) {
        $stmt = $conn->prepare('UPDATE materi SET kelas=?, judul=?, konten=?, tingkat=? WHERE id=?');
        $stmt->bind_param('ssssi', $kelas, $judul, $konten, $tingkat, $id);
        $stmt->execute();
    }
    header('Location: admin_materi.php');
    exit();
}

```

### 5.1.2 Implementasi sistem

Implementasi sistem adalah tahap penerapan sistem yang akan dilakukan jika sistem telah disetujui termasuk program yang telah dibuat pada tahap perancangan

sistem agar siap untuk dioperasikan. Adapun waktu dan tempat penerapan sistem yang sudah dibuat sebagai berikut:

Waktu dan Tempat Implementasi

Tempat: sekolah Smp Plus Multazam

Alamat: SMP Plus Multazam beralamat di Jl. Cinaggela No 73, Kecamatan Pacet, Kabupaten Bandung, Jawa Barat.

Waktu: agustus 2025

### **5.1.3 Spesifikasi sistem**

Spesifikasi sistem akan mencakup detail tentang perangkat keras dan perangkat lunak yang diperlukan untuk implementasi aplikasi pembelajaran informatika. Ini meliputi persyaratan teknis untuk server, komputer, dan perangkat yang akan digunakan. Selain itu, spesifikasi perangkat lunak akan mencakup sistem operasi, yang diperlukan untuk mendukung fungsi aplikasi. Informasi ini memastikan bahwa aplikasi dapat berjalan dengan baik dan memenuhi kebutuhan pengguna. Semua komponen ini harus sesuai agar aplikasi dapat beroperasi secara optimal.

- 1) Spesifikasi perangkat keras Dibawah ini merupakan spesifikasi perangkat keras yang digunakan sebagai berikut:

Tabel 5.1 spesifikasi perangkat keras

Ram	8Gb
Prosesor	Amd ryzen 3250
Ssd	512Gb

- 2) Spesifikasi perangkat lunak Dibawah ini merupakan spesifikasi perangkat lunak yang digunakan sebagai berikut:

## spesifikasi perangkat lunak

Sistem operasi	Windows 10
Database	MYSQL
Bahasa pemrograman	PHP

### 5.1.4 Instalasi sistem

Instalasi sistem ini memuat penjelasan mengenai langkah-langkah yang harus dilakukan untuk instalasi aplikasi dan instalasi database.

#### 1. Instalasi aplikasi

##### 1) XAMPP

XAMPP diperlukan untuk menjalankan aplikasi melalui localhost dan mengoperasikan MySQL. Berikut adalah langkah-langkah instalasi XAMPP:

- a. Unduh installer aplikasi XAMPP melalui web [apachefriends.org](http://apachefriends.org)
- b. Lakukan double klik pada file XAMPP yang sudah diunduh
- c. Klik next pada jendela installer
- d. Pilih komponen yang akan diinstal atau biarkan default untuk menginstal keseluruhan. Disarankan untuk menginstal keseluruhan
- e. Pilih folder instalasi, lalu klik next
- f. Pilih bahasa yang ingin digunakan, lalu klik next g. Jalankan instalasi dengan mengklik next pada jendela berikutnya
- g. Tunggu hingga proses instalasi selesai
- h. Setelah instalasi selesai, klik finish dan XAMPP siap digunakan

##### 2) Browser Google Chrome)

Browser ini digunakan untuk menjalankan aplikasi pembelajaran informatika. Berikut adalah langkah-langkah instalasi Google Chrome:

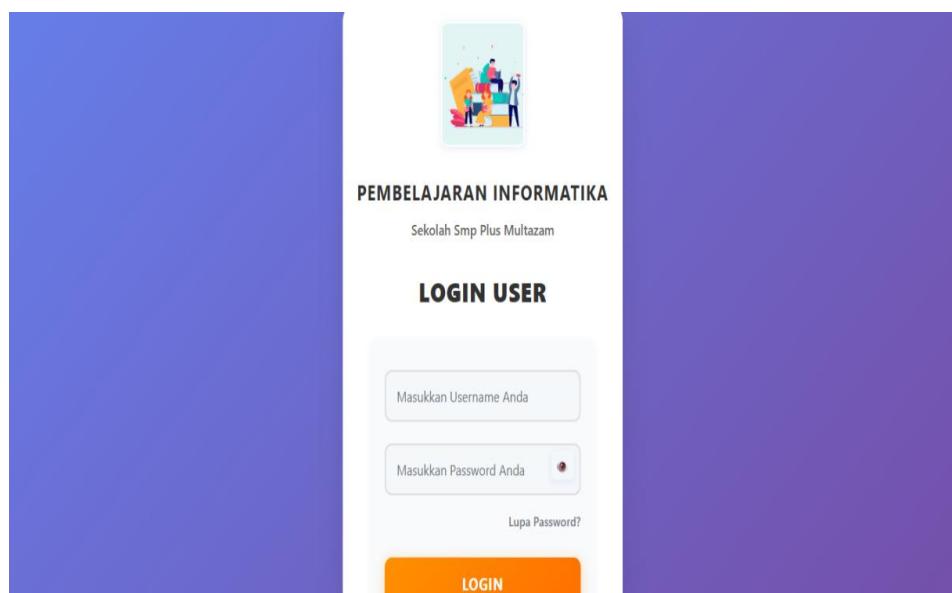
- a. Unduh installer Google Chrome
- b. Jika diminta, klik jalankan atau simpan
- c. Jika memilih simpan maka lakukan klik dua kali pada tombol download untuk memulai proses instalasi
- d. Tunggu hingga proses instalasi selesai
- e. Setelah instalasi selesai, Google Chrome sudah bisa digunakan.

Instalasi Database Database yang digunakan adalah MySQL, terintegrasi dalam XAMPPAktifkan XAMPP dan akses PHPMyAdmin di 'http://localhost/phpmyadmin' untuk mengelola database MySQL.

Pada bagian ini dijelaskan langkah-langkah dalam menjalankan sistem aplikasi Pembelajaran informatika, mulai dari proses awal hingga penggunaan fitur oleh pengguna. Penjelasan mencakup cara pengguna (siswa dan admin) mengakses aplikasi melalui browser, serta bagaimana sistem bekerja.

### 1) Login user

Fitur login user pada aplikasi pembelajaran ini berfungsi sebagai pintu masuk bagi siswa untuk mengakses materi pembelajaran secara digital. Sebelum dapat menggunakan aplikasi, siswa diwajibkan melakukan login dengan memasukkan username dan password yang telah terdaftar.



Gambar 5.1 halaman login user

pada halaman awal aplikasi, pengguna disambut dengan tampilan login user yang sederhana dan terfokus. Fitur utama halaman ini terdiri atas dua kolom input, yaitu username dan password, yang dilengkapi petunjuk untuk memandu pengguna dalam mengisi data. meningkatkan aspek kenyamanan dan keamanan. Tepat di bawahnya terdapat tautan “Lupa Password?” yang mengindikasikan dukungan terhadap pemulihan akun.

2) Halaman register

Halaman register di gunakan untuk user membuat akun baru dan bisa login ke aplikasi pembelajaran informatika.

## REGISTRASI

*Buat Akun Baru untuk Memulai Pembelajaran*

USERNAME

Masukkan username Anda

EMAIL

Masukkan email Anda

PASSWORD

Masukkan password Anda



**DAFTAR SEKARANG**

Gambar 5 1 halaman register

Halaman ini digunakan oleh pengguna baru untuk membuat akun sebelum dapat mengakses materi pembelajaran. Di bagian atas halaman, terdapat “Buat Akun Baru untuk Memulai Pembelajaran” yang menjelaskan tujuan halaman secara ringkas. Formulir registrasi terdiri dari tiga kolom input utama, yaitu Username, Email, dan Password. Masing-masing kolom dilengkapi sebagai panduan pengisian.

3) Halaman dashboard user

Halaman dashboard user menampilkan ringkasan fitur utama yang tersedia untuk pengguna setelah berhasil login.

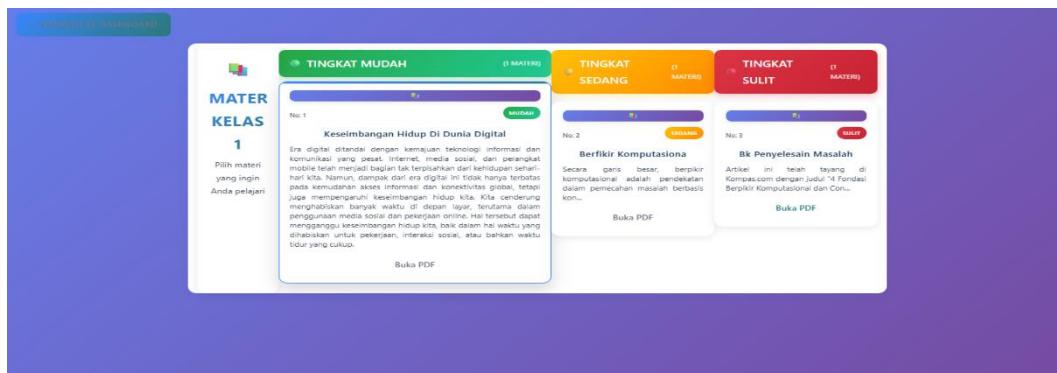


Gambar 5 2halaman dashboard user

Halaman ini merupakan tampilan utama atau beranda dari aplikasi pembelajaran yang menampilkan menu “Dashboard” di sisi kiri serta tiga menu kelas yaitu “Kelas 1”, “Kelas 2”, dan “Kelas 3” di sisi kanan Judul “Dashboard” ditampilkan secara mencolok di bagian tengah atas konten utama. Di bawahnya terdapat tiga paragraf penjelasan mengenai topik pendidikan yang mencakup: pengertian pendidikan, etimologi kata pendidikan, serta pembagian tahapan pendidikan secara umum.

4) Halaman materi kelas 1

Halaman materi kelas 1 menampilkan topik pembelajaran dasar seperti pengenalan komputer dan etika digital.

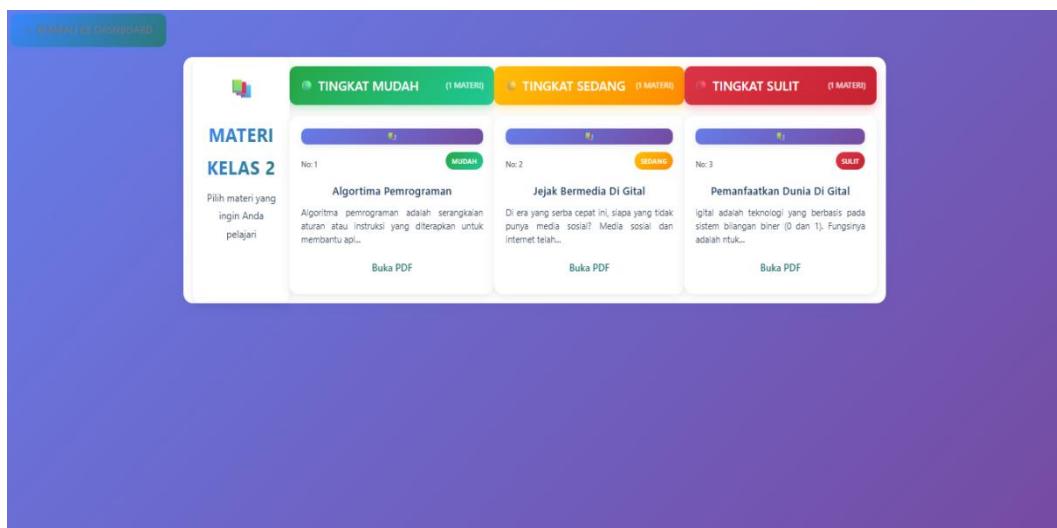


Gambar 5 3halaman materi kelas 1

Halaman ini menampilkan daftar materi pembelajaran untuk Kelas 1. Di sisi kiri terdapat panel informasi dengan judul “Materi Kelas 1” yang disertai teks pendukung “Pilih materi yang ingin Anda pelajari”, memberikan instruksi singkat kepada pengguna.

### 5) Halaman materi kelas 2

Halaman materi kelas 2 menyajikan topik lanjutan seperti perangkat lunak dan perangkat keras.

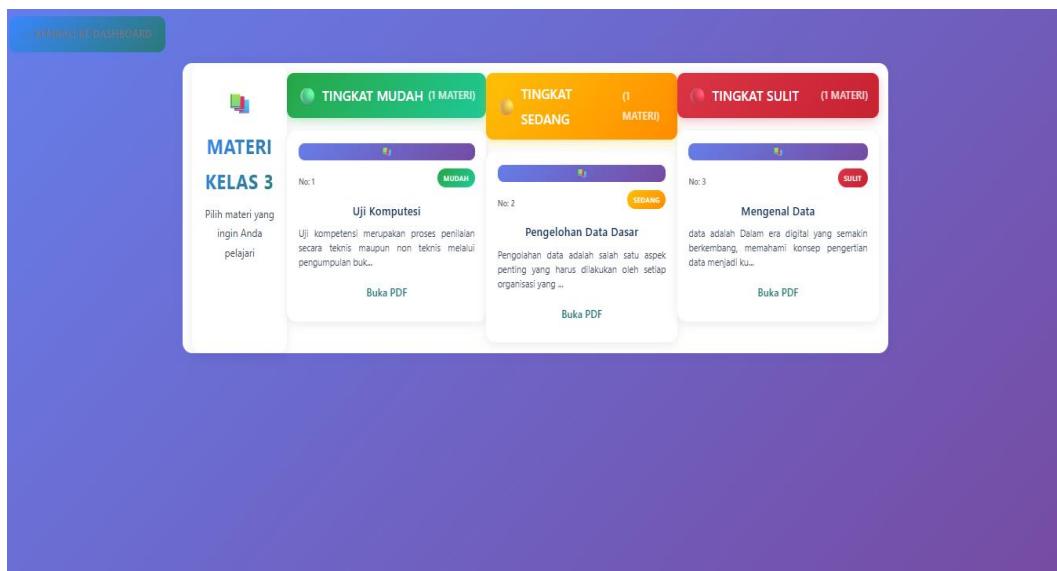


Gambar 5 4halaman materi kelas 2

Halaman Materi Kelas 2 menampilkan daftar materi pembelajaran lanjutan yang disajikan dalam tampilan bersih dan terstruktur. Di sisi kiri layar, terdapat panel informasi dengan judul “Materi Kelas 2”, disertai dengan teks “Pilih materi yang ingin Anda pelajari” yang memberikan arahan kepada pengguna. Di sisi kanan, terdapat dua kartu materi yang disusun secara vertikal. Kartu pertama berjudul “Algoritma Sederhana”, dilengkapi dengan penjelasan singkat mengenai konsep dasar algoritma beserta contoh penerapannya. Kartu kedua berjudul “Pengantar Pemrograman”, yang memuat materi pengenalan bahasa pemrograman dasar.

### 6) Halama materi kelas 3

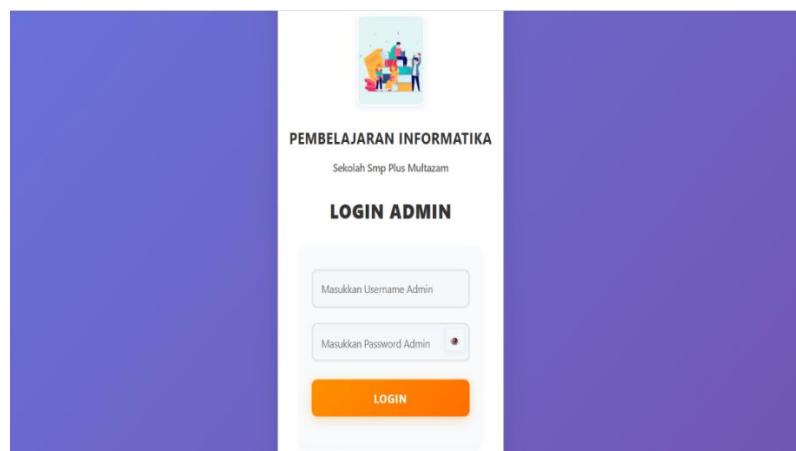
Halaman materi kelas 3 berisi topik lanjutan seperti jaringan komputer dan keamanan data.



Gambar 5 5halaman materi kelas 3

Halaman Materi Kelas 3 menampilkan daftar materi lanjutan dengan dua topik utama. Di sisi kiri, terdapat panel berjudul “Materi Kelas 3” yang dilengkapi dengan petunjuk singkat “Pilih materi yang ingin Anda pelajari”. Desain panel ini konsisten dengan kelas sebelumnya, memberikan identitas halaman yang jelas ada sisi kanan, terdapat dua kartu materi: Jaringan Komputer, yang menjelaskan konsep dasar jaringan komputer, Keamanan Data, yang membahas pentingnya menjaga data di era digital.

#### 7) Halaman login admin

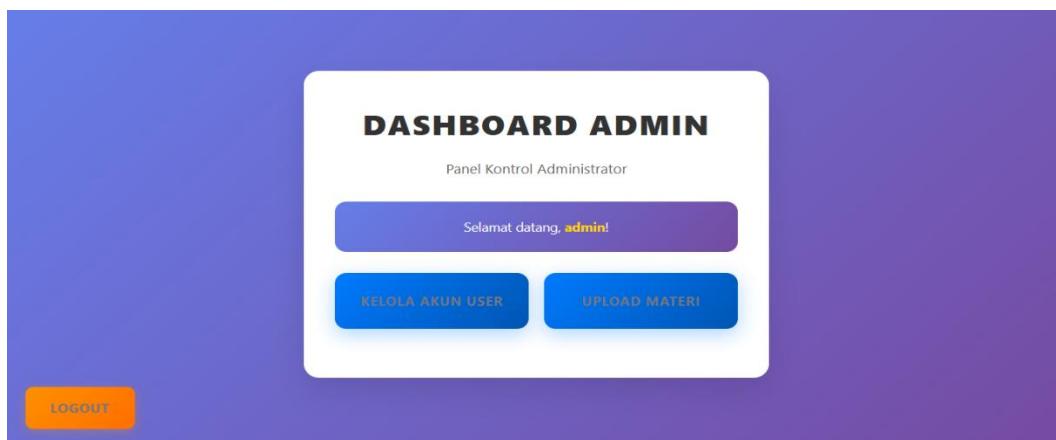


Gambar 5 6halaman login admin

Di bagian atas terdapat logo pemebelajaran yang memperkuat identitas aplikasi. Di bawahnya, terdapat formulir login yang terdiri dari dua kolom input, yaitu kolom username dan password. Pada gambar, kolom username telah diisi dengan teks “admin”, menandakan bahwa akun yang akan digunakan adalah akun administrator. Kolom password menyertakan ikon mata di sisi kanan, yang berfungsi untuk menampilkan atau menyembunyikan karakter sandi guna memudahkan pengecekan input oleh pengguna. Di bagian bawah formulir terdapat tombol aksi “sign in” berwarna biru yang digunakan untuk memproses kredensial yang dimasukkan dan mengautentikasi pengguna.

#### 8) Halaman dashbroad admin

Dashboard admin adalah halaman utama yang digunakan oleh admin untuk mengelola seluruh aktivitas dalam aplikasi.



Gambar 5 7dashbroad admin

Dashboard Admin dirancang sebagai pusat kendali bagi administrator dalam mengelola aplikasi. Di bagian tengah layar terdapat kotak utama berwarna putih dengan judul besar “DASHBOARD ADMIN” dan subjudul “Panel Kontrol Administrator”. Di bawah judul, terdapat pesan sambutan “Selamat datang, admin!” yang ditampilkan dalam gaya tulisan mencolok dengan kata “admin” diberi warna kuning terdapat dua tombol utama berwarna biru cerah, masing-masing berlabel “KELOLA AKUN USER” dan “UPLOAD MATERI”, yang berfungsi untuk mengarahkan admin ke halaman pengelolaan pengguna dan pengunggahan materi ajar.

9) Halaman kelola user admin

Halaman kelola user memungkinkan admin untuk melihat, menambah, mengedit, atau menghapus data akun pengguna.

The screenshot shows a user management interface titled "KELOLA AKUN USER". It includes a sub-section title "USER" and a subtitle "Panel Manajemen Pengguna Sistem". A button "← Kembali ke Dashboard Admin" is visible. The main area displays a table with three rows of user data:

ID	USERNAME	EMAIL	ROLE	AKSI
1	admin	admin@email.com	ADMIN	<span style="color: blue;">EDIT</span> <span style="color: red;">HAPUS</span>
2	user	user@email.com	USER	<span style="color: blue;">EDIT</span> <span style="color: red;">HAPUS</span>
3	karim	karimhasbi40@gmail.com	USER	<span style="color: blue;">EDIT</span> <span style="color: red;">HAPUS</span>

Gambar 5 8 kelola user admin

Halaman ini berfungsi sebagai panel manajemen pengguna sistem yang hanya dapat diakses oleh admin. Di sisi kiri, terdapat judul besar “KELOLA AKUN USER” disertai dengan deskripsi Aksi. Kolom Role menunjukkan status pengguna sebagai ADMIN (dengan label merah) atau USER (dengan label hijau). Di kolom aksi, terdapat dua tombol untuk setiap pengguna: Edit (berwarna biru) untuk mengubah data pengguna, dan Hapus (berwarna merah) untuk menghapus akun pengguna dari sistem.

10) Halaman kelola materi admin

Halaman kelola materi digunakan oleh admin untuk menambahkan, mengedit, atau menghapus konten pembelajaran.

The screenshot shows a lesson management interface titled "KELOLA MATERI". It includes a subtitle "Panel Manajemen Konten Pembelajaran". A button "← Kembali ke Dashboard Admin" is visible. The main area displays a table with four rows of lesson data:

ID	KELAS	JUDUL	KONTEN	TINGKAT	AKSI
1	KELAS 1	Pengenalan Komputer	Materi tentang pengenalan perangkat keras d...	MUDAH	<span style="color: blue;">EDIT</span> <span style="color: red;">HAPUS</span>
2	KELAS 1	Dasar-dasar Internet	Materi tentang apa itu internet dan cara...	MUDAH	<span style="color: blue;">EDIT</span> <span style="color: red;">HAPUS</span>
3	KELAS 2	Algoritma Sederhana	Materi tentang konsep algoritma dan contohn...	SEDANG	<span style="color: orange;">EDIT</span> <span style="color: red;">HAPUS</span>
4	KELAS 2	Pengantar Pemrograman	Materi pengenalan bahasa pemrograman...	SEDANG	<span style="color: blue;">EDIT</span> <span style="color: red;">HAPUS</span>

### Gambar 5 9 kelola materi admin

Kelola Materi berfungsi sebagai panel manajemen konten pembelajaran bagi admin. Di sisi kiri layar, terdapat judul besar “KELOLA MATERI” dengan subjudul “Panel Manajemen Konten Pembelajaran”. Di bawahnya, tersedia dua tombol navigasi, yaitu tombol abu-abu bertuliskan “← Kembali ke Dashboard Admin” untuk kembali ke halaman utama admin, serta tombol hijau “+ Tambah Materi Baru” untuk menambahkan materi pembelajaran baru.

## **5.2 Pengujian aplikasi**

Tahap berikutnya setelah proses perancangan dan pengembangan adalah melakukan pengujian terhadap aplikasi. Pengujian ini merupakan langkah penting dalam memastikan bahwa seluruh fitur dan fungsi yang telah dibangun dapat berjalan dengan baik sesuai dengan tujuan awal pengembangan. Melalui pengujian, dapat diketahui apakah aplikasi telah memenuhi kebutuhan pengguna, apakah terdapat bug atau kesalahan dalam sistem, serta bagaimana performa aplikasi ketika digunakan dalam berbagai skenario. Berikut pengujian yang dilakukan:

### **5.2.1 black box testing**

Tahap berikutnya adalah melakukan pengujian terhadap aplikasi. Pada tahap ini, dilakukan pengujian menggunakan metode Black Box Testing guna memastikan bahwa fungsi-fungsi dalam aplikasi telah berjalan sesuai dengan perancangan.

#### 1) User

Tabel pengujian black box untuk user

No.	Item	Skenario pengujian	Hasil yang diharapkan	Hasil pengujian
1.	Login	Siswa mengisi username dan password	Siswa dapat masuk ke halaman dashbroad	SESUAI
2.	Register	Bila belum mempunyai akun	Maka sistem akan	SESUAI

		siswa dapat membuat akun baru	memunculkan halaman register	
3.	Dashbroad	Siswa masuk ke halaman dashboard	Sistem akan memunculkan halaman dashbroad	SESUAI
4.	Kelas 1	Siswa yang kelas 1 memilih kelas 1	Sistem akan memunculkan materi ajar kelas 1	SESUAI
5.	Kelas 2	Siswa kelas 2 memilih kelas 2	Sistem akan memunculkan materi ajar kelas 2	SESUAI
6.	Kelas 3	Siswa kelas 3 memilih kelas 3	Sistem akan memunculkan materi ajar kelas 3	SESUAI

2) Admin

No.	Item	Skenario pengujian	Hasil yang diharapkan	Hasil pengujian
1.	Login	Admin mengisi username dan password	Admin berhasil dan masuk ke halaman dashbroad	SESUAI
2.	Kelola akun user	Admin dapat mengedit atau menghapus akun user	Dan data siswa berhasil di perbarui	SESUAI

3.	Upload materi	Admin dapat mengupload materi dan bisa mengedit atau hapus materi	Dan data berhasil ditambahkan, diperbarui, atau dihapus sesuai aksi	SESUAI
----	---------------	---	---	--------

Berdasarkan pengujian yang ditampilkan pada Tabel 5.3 dan Tabel 5.4 yang memuat hasil pengujian aplikasi dari sisi siswa (user) dan admin, dapat disimpulkan bahwa aplikasi pembelajaran informatika telah berfungsi sesuai dengan hasil analisis dan perancangan sistem. Seluruh fitur yang dikembangkan mampu berjalan dengan baik dan sesuai dengan tujuan fungsional yang telah ditetapkan sebelumnya

## BAB VI

### KESIMPULAN DAN SARAN

#### 6.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan melalui tahapan analisis kebutuhan, perancangan sistem, implementasi aplikasi, serta pengujian pada bab-bab sebelumnya, maka dapat disimpulkan bahwa:

1. Aplikasi pembelajaran berbasis web yang dikembangkan berhasil membantu proses belajar mengajar mata pelajaran Informatika di SMP Plus Multazam secara lebih interaktif, fleksibel, dan sesuai dengan kurikulum yang berlaku. Aplikasi ini mendukung akses pembelajaran kapan saja dan di mana saja melalui perangkat desktop.
2. Penggunaan algoritma *Quick Sort* dalam sistem terbukti efektif dalam mengurutkan materi pembelajaran berdasarkan tingkat kesulitan dan relevansi. Hal ini memudahkan siswa untuk mengakses materi secara bertahap dan terstruktur.
3. Fitur-fitur utama yang tersedia, seperti halaman login, dashboard, pembelajaran per kelas, serta logout, mampu memenuhi kebutuhan pengguna (guru dan siswa). Dashboard menjadi pusat kontrol yang menyajikan informasi dan navigasi yang sesuai dengan peran masing-masing pengguna.
4. Hasil pengujian menggunakan metode black-box testing menunjukkan bahwa seluruh fitur utama dalam aplikasi dapat berjalan dengan baik dan sesuai fungsinya. Aplikasi dinilai mudah digunakan oleh siswa maupun guru berdasarkan pengujian internal.
5. Penerapan metode Waterfall dalam pengembangan sistem memberikan struktur yang sistematis dan terencana. Tahapan mulai dari analisis hingga pemeliharaan dapat dilaksanakan dengan runtut dan terarah, menghasilkan sistem yang stabil dan sesuai dengan kebutuhan pengguna.

## 6.2 Saran

Berdasarkan hasil penelitian dan pengembangan aplikasi ini, penulis memberikan beberapa saran untuk pengembangan lebih lanjut:

1. Pengembangan antarmuka mobile-friendly agar aplikasi dapat diakses secara optimal melalui perangkat seluler, sehingga menjangkau lebih banyak pengguna.
2. Penambahan fitur evaluasi otomatis, seperti kuis atau soal latihan yang terintegrasi dengan sistem penilaian untuk mendukung proses pembelajaran yang lebih komprehensif.
3. Integrasi multimedia seperti video pembelajaran, animasi interaktif, atau simulasi dapat ditambahkan agar materi menjadi lebih menarik dan mudah dipahami.
4. Peningkatan keamanan sistem, seperti penggunaan fitur CAPTCHA, autentifikasi dua langkah (2FA), dan fitur “lupa password” agar mencegah akses tidak sah ke dalam sistem.
5. Dashboard analitik untuk guru, yang menampilkan data perkembangan belajar siswa, agar guru dapat memantau efektivitas pembelajaran dan memberikan umpan balik secara tepat.
6. Pengujian pengguna eksternal dari luar sekolah SMP Plus Multazam juga direkomendasikan untuk menguji skalabilitas dan adaptabilitas sistem terhadap institusi pendidikan lainnya.

## DAFTAR PUSTAKA

- Bambang Kurnia, & Tumini Tumini. (2023). Perancangan Sistem Informasi Pembelajaran Online (E-Learning) Berbasis Web. *Jurnal Ilmiah Teknik Informatika Dan Komunikasi*, 3(3), 276–286.  
<https://doi.org/10.55606/juitik.v3i3.714>
- Chandrawati, W. M., Rusmawati, R. D., & Harwanto, H. (2023). Pengembangan E-Learning Berbasis Media Interaktif Smart Apps Creator terhadap Motivasi Belajar Siswa SMP Empat Lima 2 Kedungpring pada Pelajaran Informatika. *JIIP - Jurnal Ilmiah Ilmu Pendidikan*, 6(11), 9146–9154.  
<https://doi.org/10.54371/jiip.v6i11.2941>
- DAUM, B. (2003). XML Basics. *Modeling Business Objects with XML Schema*, November, 89–115. <https://doi.org/10.1016/b978-155860816-0/50006-9>
- Hardiyana, B., & Nopandi, Y. (2014). Rancang Bangun Aplikasi Pembelajaran Pemrograman Berorientasi Objek Dengan Bahasa Java. *Prodi Sistem Informasi Universitas Komputer Indonesia*.
- Harry Saptarini, N. G. A. P., Hidayat, R. A., & Ciptayani, P. I. (2019). Ajarincode : Aplikasi Pembelajaran Bahasa Pemrograman Berbasis Web. *Just TI (Jurnal Sains Terapan Teknologi Informasi)*, 10(2), 21.  
<https://doi.org/10.46964/justti.v10i2.106>
- Kasus, S., Sistem, A., Piutang, M., & Panjang, J. (2025). *Implementasi Algoritma Quick Sort Pada Aplikasi Pemrograman Berorientasi Objek Berbasis Python Dan Mysql Workbench*. 14(164), 90–110.
- Mata, P. E., Informatika, P., Bukittinggi, D. I. S., Pangaribuan, F., Zakir, S., Efriyanti, L., Studi, P., Teknik, P., & Tarbiyah, F. (2023). *Berbasis Kurikulum Merdeka Menggunakan Efront*. 7(3).
- Mukti, I. S. ., Lumenta, A. S. ., & Sugiarso, B. A. (2016). Rancang Bangun Aplikasi Pembelajaran Untuk. *E-Journal Teknik Informatika*, 7(1), 1–6.
- Prasetio, I., & Musril, H. A. (2022). Perancangan Media Pembelajaran Fisika Berbasis Android Menggunakan Smart Apps Creator 3. *Jurnal Manajemen*

- Informatika (Jumika), 8(2), 1–15. <https://doi.org/10.51530/jumika.v8i2.546>*
- Rosdiana, D. (2024). *Implementasi Algoritma Quick Sort dalam Sistem Pemesanan Makanan di Rumah Makan.* 1(2), 60–65.
- Siame, A., & Kunda, D. (2017). Evolution of PHP Applications: A Systematic Literature Review. *International Journal of Recent Contributions from Engineering, Science & IT (IJES)*, 5(1), 28. <https://doi.org/10.3991/ijes.v5i1.6437>
- Susanto, W. E., & Ayu, Y. G. A. (2017). Perancangan E-Learning Berbasis Web Pada SMP Negeri 3 Patuk Gunungkidul Yogyakarta. *Bianglala Informatika*, 5(2).
- Šušter, I., & Ranisavljević, T. (2023). Optimization of MySQL database. *Journal of Process Management and New Technologies*, 11(1–2), 141–151. <https://doi.org/10.5937/jouproman2301141q>
- Tan, J., Chen, Y., & Jiao, S. (2024). Visual Studio Code in Introductory Computer Science Course: An Experience Report. In *ASEE Annual Conference and Exposition, Conference Proceedings* (Vol. 1, Issue 1). Association for Computing Machinery. <https://doi.org/10.18260/1-2--48259>
- Tuti, H. (2022a). Rancang Bangun Aplikasi Pembelajaran Bangun Ruang Berbasis Unity. *Jurnal Teknologi Technoscientia*, 16, 109–120. <https://doi.org/10.34151/technoscientia.v14i2.3628>
- Tuti, H. (2022b). Rancang Bangun Aplikasi Pembelajaran Bangun Ruang Berbasis Unity. *Jurnal Teknologi Technoscientia*, 109–120. <https://doi.org/10.34151/technoscientia.v14i2.3628>

## LAMPIRAN

### Lampiran 1: Hasil wawancara dengan pihak sekolah

Narasember :Lilis Rohaeti, S.Pd  
Jabatan/ posisi : Guru  
Hari/tanggal : 19 mei 2025  
Intansi : Smp Plus Multazam

Wawancara ini bertujuan untuk memperoleh data penelitian terkait implementasi aplikasi pembelajaran informatika di Smp Plus Multazam.

Berikut merupakan rangkaian pertanyaan wawancara dan hasil jawabannya.

No	Pertanyaan	Jawaban
1	Bagaimana metode pembelajaran Informatika yang saat ini digunakan di sekolah?	Pembelajaran masih menggunakan metode konvensional seperti ceramah dan buku cetak, dengan sedikit praktik komputer di laboratorium.
2	Apa kendala yang dihadapi guru dalam menyampaikan materi Informatika?	Keterbatasan media pembelajaran interaktif, sulit memantau perkembangan siswa secara detail, dan kurangnya akses materi di luar jam pelajaran.
3	Apakah sekolah sudah memiliki sistem pembelajaran berbasis web?	Saat ini belum ada sistem khusus pembelajaran berbasis web yang terintegrasi untuk mata pelajaran Informatika.
4	Fitur apa saja yang diharapkan ada dalam aplikasi pembelajaran berbasis web?	Akses materi sesuai kurikulum, latihan soal, forum diskusi, sistem penilaian, dan kemampuan mengurutkan materi berdasarkan tingkat kesulitan.
5	Bagaimana pandangan sekolah terhadap penggunaan	Mendukung, karena dapat membantu mengurutkan materi secara otomatis dan

	algoritma <i>Quick Sort</i> pada sistem?	memudahkan siswa belajar sesuai level kemampuan mereka.
6	Apakah guru siap untuk menggunakan dan mengelola aplikasi ini?	Siap, asalkan diberikan pelatihan terlebih dahulu untuk memahami cara penggunaan dan pengelolaan sistem.
7	Bagaimana harapan sekolah terhadap penerapan aplikasi ini di masa depan?	Diharapkan dapat meningkatkan minat dan pemahaman siswa terhadap materi Informatika, serta mempermudah guru dalam menyampaikan materi.

Pewawancara

Narasumber

Hasbi Kurnia Karim

Lilis Rohaeti, S.Pd

## Lampiran 2: Dokumentasi Wawancara

Nama instansi: Smp Plus Multazam

Lokasi : Jl. Cinaggela No 73, Kecamatan Pacet, Kabupaten Bandung, Jawa Barat.



### Lampiran 3: Listing program

#### Login User

```
<?php
session_start();
if (isset($_SESSION['user_id'])) {
    header('Location: dashboard.php');
    exit();
}
$error = $_SESSION['login_error'] ?? '';
unset($_SESSION['login_error']);
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Login - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>
</head>
<body class="login-bg">
    <div class="login-card centered">
        
        <div class="school-title">
            <div>Pembelajaran informatika</div>
            <div class="school-name">Sekolah Smp Plus
Multazam</div>
        </div>
        <div class="login-title">Login User</div>
        <?php if ($error): ?>
            <div class="error"> <?=
htmlspecialchars($error) ?> </div>
        <?php endif; ?>
```

```

        <form action="login_process.php"
method="post">
            <input type="text" name="username"
placeholder="Masukkan Username Anda" required>
            <div class="password-container">
                <input type="password" name="password"
id="password" placeholder="Masukkan Password Anda"
required>
                <button type="button" class="password-
toggle" onclick="togglePassword()">👁</button>
            </div>
            <a href="forgot_password.php"
class="forgot-link">Lupa Password?</a>
            <button type="submit" class="login-
btn">Login</button>
        </form>
        <div class="register-section" style="margin-
top: 20px; text-align: center; padding: 20px; background:
rgba(255,255,255,0.98); border-radius: 12px; border: 3px
solid rgba(102,126,234,0.6); box-shadow: 0 8px 25px
rgba(0,0,0,0.2);">
            <p style="color: #333; font-size: 1rem;
margin-bottom: 18px; font-weight: 700; text-transform:
uppercase; letter-spacing: 0.5px;">📝 Belum punya
akun?</p>
            <a href="register.php" class="register-
link" style="display: inline-block !important; visibility:
visible !important; opacity: 1 !important; font-size:
1.1rem !important; padding: 15px 30px !important;">🚀
Daftar Sekarang</a>
        </div>
    </div>

<script>
    function togglePassword() {

```

```

        const passwordInput =
document.getElementById('password');

        const toggleButton =
document.querySelector('.password-toggle');

        if (passwordInput.type === 'password') {
            passwordInput.type = 'text';
            toggleButton.textContent = '🙈';
        } else {
            passwordInput.type = 'password';
            toggleButton.textContent = '👁';
        }
    
```

</script>

</body>

</html>

## **Listing Program Refgister**

```

<?php
session_start();
if (isset($_SESSION['user_id'])) {
    header('Location: dashboard.php');
    exit();
}
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Registrasi - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>

```

```

</style>
</head>
<body>

    <div class="register-container">
        <div class="register-header">
            <h1 class="register-title">Registrasi</h1>
            <p class="register-subtitle">Buat Akun Baru  
untuk Memulai Pembelajaran</p>
        </div>

        <form action="register_process.php" method="post">
            <div class="form-group">
                <label for="username">Username</label>
                <input type="text" id="username" name="username" placeholder="Masukkan username Anda" required>
            </div>

            <div class="form-group">
                <label for="email">Email</label>
                <input type="email" id="email" name="email" placeholder="Masukkan email Anda" required>
            </div>

            <div class="form-group">
                <label for="password">Password</label>
                <div class="password-container">
                    <input type="password" id="password" name="password" placeholder="Masukkan password Anda" required>
                    <button type="button" class="password-toggle" onclick="togglePassword()">@</button>
                </div>
            </div>
        </form>
    </div>

```

```

        <button type="submit" class="register-btn">Daftar Sekarang</button>
    </form>

        <a href="login.php" class="login-link">Sudah punya akun? Login di sini</a>
    </div>

<script>
    function togglePassword() {
        const passwordInput = document.getElementById('password');
        const toggleBtn = document.querySelector('.password-toggle');

        if (passwordInput.type === 'password') {
            passwordInput.type = 'text';
            toggleBtn.textContent = '🙈';
        } else {
            passwordInput.type = 'password';
            toggleBtn.textContent = '👁';
        }
    }
</script>
</body>
</html>

```

### **Listing Program Forget password**

```

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Lupa Password - Pembelajaran Informatika</title>
<link rel="stylesheet" href="assets/style.css">
</head>
<body>
    <h2>Lupa Password</h2>
    <form action="forgot_password_process.php" method="post">
        <label>Email:</label><br>
        <input type="email" name="email" required><br>
        <button type="submit">Kirim Link Reset</button>
    </form>
    <a href="login.php">Kembali ke Login</a>
</body>
</html>

```

### **Listing Program Materi**

```

<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

include 'includes/db.php';

// Cek apakah ada parameter id
if (!isset($_GET['id']) || empty($_GET['id'])) {
    header('Location: dashboard.php');
    exit();
}

$id = mysqli_real_escape_string($conn, $_GET['id']);
$query = "SELECT * FROM materi WHERE id = '$id'";
$result = mysqli_query($conn, $query);

```

```

// Cek apakah query berhasil dan ada data
if (!$result || mysqli_num_rows($result) == 0) {
    echo "<script>alert('Materi tidak ditemukan!');  

window.location.href='dashboard.php';</script>";
    exit();
}

$materi = mysqli_fetch_assoc($result);
// Tambahkan pengecekan jika data tidak ditemukan atau
konten kosong
if (!$materi || empty($materi['konten'])) {
    echo '<div style="max-width:600px;margin:40px  

auto;padding:30px;background:#fff;border-radius:10px;text-  

align:center;box-shadow:0 2px 10px rgba(0,0,0,0.08);">';
    echo '<h2>Materi tidak ditemukan atau belum memiliki  

konten.</h2>';
    echo '<a href="dashboard.php" style="display:inline-  

block;margin-top:20px;padding:10px  

24px;background:#007bff;color:#fff;border-radius:6px;text-  

decoration:none;">Kembali ke Dashboard</a>';
    echo '</div>';
    exit();
}
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,  

initial-scale=1.0">
    <title><?php echo htmlspecialchars($materi['judul']);  

?> - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">

```

```
<style>
```

## Listing program Kelas 1

```
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

include 'includes/db.php';

// Ambil materi untuk kelas 1
$query = "SELECT * FROM materi WHERE kelas = 1 ORDER BY id";
$result = mysqli_query($conn, $query);
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Kelas 1 - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>

</head>
<body>
    <a href="dashboard.php" class="back-btn">← Kembali ke
Dashboard</a>

    <div class="container">
        <div class="page-header">
```

```

        <h1 class="page-title">Materi Kelas 1</h1>
        <p class="page-subtitle">Pilih materi yang ingin
Anda pelajari</p>
        </div>

        <div class="materi-grid">
            <?php if (mysqli_num_rows($result) > 0): ?>
                <?php while ($materi =
mysqli_fetch_assoc($result)): ?>
                    <div class="materi-card"
onclick="window.location.href='materi_detail.php?id=<?php
echo $materi['id']; ?>'>
                        <?php if (!empty($materi['foto'])):
?>
                            
                        <?php else: ?>
                            <div class="materi-img"
style="background: linear-gradient(135deg, #667eea 0%,
#764ba2 100%); display: flex; align-items: center; justify-
content: center; color: white; font-weight: bold; font-
size: 18px;">
                                <img alt="Placeholder image" style="width: 40px; height: 40px; border-radius: 50%; margin-right: 10px;" />
                            </div>
                        <?php endif; ?>
                    <div class="materi-title"><?php echo
htmlspecialchars($materi['judul']); ?></div>
                    <div class="materi-desc">
                        <?php
                            echo
htmlspecialchars(mb_strimwidth($materi['konten'], 0, 100,
'...'));
                        ?>
                    </div>

```

```

        </div>
    <?php endwhile; ?>
    <?php else: ?>
        <div class="materi-card empty-state">
            <div class="materi-title">Belum ada
materi</div>
            <div class="materi-desc">Materi untuk
kelas 1 akan segera ditambahkan.</div>
        </div>
    <?php endif; ?>
</div>
</div>
</body>
</html>

```

## **Listing program kelas 2**

```

<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

include 'includes/db.php';

// Ambil materi untuk kelas 2
$query = "SELECT * FROM materi WHERE kelas = 2 ORDER BY id";
$result = mysqli_query($conn, $query);
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">

```

```

        <meta name="viewport" content="width=device-width,
initial-scale=1.0">

        <title>Kelas 2 - Pembelajaran Informatika</title>
        <link rel="stylesheet" href="assets/style.css">
        <style>
</style>
</head>
<body>

        <a href="dashboard.php" class="back-btn">← Kembali ke
Dashboard</a>

        <div class="container">
            <div class="page-header">
                <h1 class="page-title">Materi Kelas 2</h1>
                <p class="page-subtitle">Pilih materi yang ingin
Anda pelajari</p>
            </div>

            <div class="materi-grid">
                <?php if (mysqli_num_rows($result) > 0): ?>
                    <?php while ($materi =
mysqli_fetch_assoc($result)): ?>
                        <div class="materi-card"
onclick="window.location.href='materi_detail.php?id=<?php
echo $materi['id']; ?>'>
                            <div class="materi-img"
style="background: linear-gradient(135deg, #667eea 0%,
#764ba2 100%); display: flex; align-items: center; justify-
content: center; color: white; font-weight: bold; font-
size: 18px; width: 100%; max-height: 180px; object-fit:
cover; border-radius: 8px; margin-bottom: 12px;">
                                <img alt="Material design card icon" style="width: 40px; height: 40px; border-radius: 50%; background-color: #fff; border: 1px solid #ccc; margin-right: 10px;" />
                            </div>
                            <div class="materi-title"><?php echo
htmlspecialchars($materi['judul']); ?></div>
                        </div>
                <?php endwhile; ?>
            </div>
        </div>
    </body>

```

```

<div class="materi-desc">
    <?php
        echo
        htmlspecialchars(mb_strimwidth($materi['konten'], 0, 100,
        '...'));
    ?>
</div>
</div>
<?php endwhile; ?>
<?php else: ?>
    <div class="materi-card empty-state">
        <div class="materi-title">Belum ada
        materi</div>
        <div class="materi-desc">Materi untuk
        kelas 2 akan segera ditambahkan.</div>
    </div>
    <?php endif; ?>
</div>
</div>
</body>
</html>

```

### **Listing Program Kelas 3**

```

<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit();
}

include 'includes/db.php';

// Ambil materi untuk kelas 3
$query = "SELECT * FROM materi WHERE kelas = 3 ORDER BY id";

```

```

$result = mysqli_query($conn, $query);

?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Kelas 3 - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>

        </style>
</head>
<body>
    <a href="dashboard.php" class="back-btn">← Kembali ke
Dashboard</a>

    <div class="container">
        <div class="page-header">
            <h1 class="page-title">Materi Kelas 3</h1>
            <p class="page-subtitle">Pilih materi yang ingin
Anda pelajari</p>
        </div>

        <div class="materi-grid">
            <?php if (mysqli_num_rows($result) > 0): ?>
                <?php while ($materi =
mysqli_fetch_assoc($result)): ?>
                    <div class="materi-card"
onclick="window.location.href='materi_detail.php?id=<?php
echo $materi['id']; ?>'>
                        <div class="materi-img"
style="background: linear-gradient(135deg, #667eea 0%,",

```

```

#764ba2 100%); display: flex; align-items: center; justify-
content: center; color: white; font-weight: bold; font-
size: 18px; width: 100%; max-height: 180px; object-fit:
cover; border-radius: 8px; margin-bottom: 12px;">

</div>
<div class="materi-title"><?php echo
htmlspecialchars($materi['judul']); ?></div>
<div class="materi-desc">
<?php
echo
htmlspecialchars(mb_strimwidth($materi['konten'], 0, 100,
'...'));
?>
</div>
</div>
<?php endwhile; ?>
<?php else: ?>
<div class="materi-card empty-state">
<div class="materi-title">Belum ada
materi</div>
<div class="materi-desc">Materi untuk
kelas 3 akan segera ditambahkan.</div>
</div>
<?php endif; ?>
</div>
</div>
</body>
</html>

```

### **Listing program Dashbroad user**

```

DOCTYPE html>
<html lang="id">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Dashboard - Pembelajaran Informatika</title>
<style>
</style>
</head>
<body>

<div class="header">
<div class="nav">
<a href="#" class="nav-dashboard">Dashboard</a>
<div class="nav-right">
<a href="kelas1.php">Kelas 1</a>
<a href="kelas2.php">Kelas 2</a>
<a href="kelas3.php">Kelas 3</a>
</div>
</div>
</div>

<div class="content" style="background:#e6f7fa; display:flex; align-items:center; justify-content:center; min-height:400px; padding:40px 0;">
<div style="flex:0 0 380px; display:flex; justify-content:center; align-items:center;">

</div>
<div style="flex:1; max-width:700px; margin-left:40px;">
<h1 style="text-align:center; font-size:2.8rem; font-family:'Montserrat',Arial,sans-serif; font-weight:900; margin-bottom:28px; letter-spacing:1px;">Dashboard</h1>
<p style="font-size:1.1rem; line-height:1.7; text-align:justify; margin:0;">
Pendidikan adalah pembelajaran pengetahuan, keterampilan, dan kebiasaan sekelompok orang yang diturunkan dari satu generasi ke generasi berikutnya melalui pengajaran, pelatihan, atau penelitian. Pendidikan sering terjadi di bawah bimbingan orang lain, tetapi juga memungkinkan secara otodidak.<br><br>
Etimologi kata pendidikan itu sendiri berasal dari bahasa Latin yaitu ducere, berarti “menuntun, mengarahkan, atau memimpin” dan awalan e, berarti “keluar”. Jadi,

```

pendidikan berarti kegiatan “menuntun ke luar”. Setiap pengalaman yang memiliki efek formatif pada cara orang berpikir, merasa, atau tindakan dapat dianggap pendidikan.<br><br>

Pendidikan umumnya dibagi menjadi tahap seperti prasekolah, sekolah dasar, sekolah menengah pertama, sekolah menengah atas, dan kemudian perguruan tinggi, universitas atau magang.

```
</p>
</div>
</div>

<div class="footer">
</div>
<a href="logout.php" class="logout-fixed">Logout</a>

</body>
</html>
```

### **Listing program logout**

```
<?php
session_start();
session_unset();
session_destroy();
header('Location: login.php');
exit();
?>
```

### **Listing program admin login**

```
<?php
session_start();
if (isset($_SESSION['admin_id'])) {
    header('Location: admin_dashboard.php');
    exit();
}
```

```

}

$error = $_SESSION['admin_login_error'] ?? '';
unset($_SESSION['admin_login_error']);
?>
<!DOCTYPE html>
<html lang="id">
<head>
<meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Login Admin - Pembelajaran Informatika</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>
</style>
</head>
<body class="login-bg">
    <div class="login-card centered">
        
        <div class="school-title">
            <div>Pembelajaran informatika</div>
            <div class="school-name">Sekolah Smp Plus
Multazam</div>
        </div>
        <div class="login-title">Login Admin</div>
        <?php if ($error) : ?>
            <div class="error"> <?= htmlspecialchars($error)
?> </div>
        <?php endif; ?>
        <form action="admin_login_process.php" method="post">
            <input type="text" name="username"
placeholder="Masukkan Username Admin" required>
            <div class="password-container">

```

```

                <input type="password" name="password"
id="password" placeholder="Masukkan Password Admin"
required>

                <button type="button" class="password-toggle"
onclick="togglePassword()">❸</button>
            </div>
            <button type="submit" class="login-
btn">Login</button>
        </form>
        <a href="index.php" class="back-link">Kembali ke
Beranda</a>
    </div>

    <script>
        function togglePassword() {
            const passwordInput =
document.getElementById('password');
            const toggleButton =
document.querySelector('.password-toggle');

            if (passwordInput.type === 'password') {
                passwordInput.type = 'text';
                toggleButton.textContent = '❹';
            } else {
                passwordInput.type = 'password';
                toggleButton.textContent = '❺';
            }
        }
    </script>
</body>
</html>

```

### **Listing program Dashbroad admin**

```

<?php
session_start();
if (!isset($_SESSION['admin_id'])) {

```

```

        header('Location: admin_login.php');
        exit();
    }
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0">
            <title>Dashboard Admin - Pembelajaran Informatika</title>
            <link rel="stylesheet" href="assets/style.css">
            <style>

</style>
</head>
<body>
    <div class="admin-container">
        <div class="admin-header">
            <h1 class="admin-title">Dashboard Admin</h1>
            <p class="admin-subtitle">Panel Kontrol Administrator</p>
        </div>

        <div class="admin-welcome">
            Selamat datang, <b><?php echo htmlspecialchars($_SESSION['admin_username']); ?></b>!
        </div>

        <div class="admin-menu">
            <a href="admin_users.php" class="admin-btn">Kelola Akun User</a>
            <a href="admin_materi.php" class="admin-btn">Upload Materi</a>
        </div>
    </div>

```

```

        </div>
    </div>

        <a href="admin_logout.php" class="logout-admin">Logout</a>
    </body>
</html>

```

### **Listing program admin materi edit**

```

<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';
$id = $_GET['id'] ?? '';
if ($_SERVER['REQUEST_METHOD'] === 'POST' && $id) {
    $kelas = $_POST['kelas'] ?? '';
    $judul = $_POST['judul'] ?? '';
    $konten = $_POST['konten'] ?? '';
    $tingkat = $_POST['tingkat'] ?? 'mudah';
    if ($kelas && $judul && $konten && $tingkat) {
        $stmt = $conn->prepare('UPDATE materi SET kelas=?, judul=?, konten=?, tingkat=? WHERE id=?');
        $stmt->bind_param('ssssi', $kelas, $judul, $konten, $tingkat, $id);
        $stmt->execute();
    }
    header('Location: admin_materi.php');
    exit();
}
?>

```

### **Listing program admin materi delet**

```
<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';
$id = $_GET['id'] ?? '';
if ($id) {
    $stmt = $conn->prepare('DELETE FROM materi WHERE id=?');
    $stmt->bind_param('i', $id);
    $stmt->execute();
}
header('Location: admin_materi.php');
exit();
?>
```

### **Listing program admin upload materi**

```
<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $kelas = $_POST['kelas'] ?? '';
    $judul = $_POST['judul'] ?? '';
    $konten = $_POST['konten'] ?? '';
    $tingkat = $_POST['tingkat'] ?? 'mudah';
    if ($kelas && $judul && $konten && $tingkat) {
```

```

$stmt = $conn->prepare('INSERT INTO materi (kelas,
judul, konten, tingkat) VALUES (?, ?, ?, ?)');
$stmt->bind_param('ssss', $kelas, $judul, $konten,
$tingkat);
$stmt->execute();
}
header('Location: admin_materi.php');
exit();
}
?>

```

### **Listing program materi kelola admin**

```

_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';
$materi = $conn->query("SELECT * FROM materi ORDER BY kelas,
created_at DESC");
?>
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Kelola Materi - Admin</title>
    <link rel="stylesheet" href="assets/style.css">
    <style>
</style>
</head>
<body>
    <div class="container">

```

```

<div class="page-header">
    <h1 class="page-title">Kelola Materi</h1>
    <p class="page-subtitle">Panel Manajemen Konten
Pembelajaran</p>
</div>

<div class="nav-links">
    <div class="nav-left">
        <a href="admin_dashboard.php" class="back-
link">← Kembali ke Dashboard Admin</a>
    </div>
    <div class="nav-right">
        <a href="admin_materi_add.php" class="add-
link">+ Tambah Materi Baru</a>
    </div>
</div>

<div class="table-container">
<table>
    <thead>
        <tr>
            <th>ID</th>
            <th>Kelas</th>
            <th>Judul</th>
            <th>Konten</th>
            <th>Tingkat</th>
            <th>Aksi</th>
        </tr>
    </thead>
    <tbody>
<?php while ($m = $materi->fetch_assoc()): ?>
<tr>
        <td class="id-cell"><?= $m['id'] ?></td>

```

```

        <td><span class="kelas-badge">Kelas
<?= $m['kelas'] ?></span></td>
                <td class="judul-cell"><?=
htmlspecialchars($m['judul']) ?></td>
                <td>
                        <div class="konten-preview">
                                <?= nl2br(htmlspecialchars($m['konten'])) ?>
                        </div>
                </td>
                <td>
                        <span class="tingkat-badge"
tingkat-<?= $m['tingkat'] ?>">
                                <?= ucfirst($m['tingkat']) ?>
                ?>
                        </span>
                </td>
                <td class="action-links">
                        <a href="admin_materi_edit.php?id=<?=$m['id'] ?>" class="edit-link" title="Edit Materi">Edit</a>
                        <a href="admin_materi_delete.php?id=<?=$m['id'] ?>" onclick="return confirm('Yakin hapus materi ini?')" class="delete-link" title="Hapus Materi">Hapus</a>
                </td>
        </tr>
        <?php endwhile; ?>
    </tbody>
</table>
</div>
</div>
</body>

```

```
</html>
```

### Listing program admin keolala user

```
<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $username = $_POST['username'] ?? '';
    $email = $_POST['email'] ?? '';
    $password = $_POST['password'] ?? '';
    $role = $_POST['role'] ?? 'user';

    if ($username && $email && $password && $role) {
        // Cek apakah username sudah digunakan
        $stmt = $conn->prepare('SELECT id FROM users WHERE
username = ?');
        $stmt->bind_param('s', $username);
        $stmt->execute();
        $result = $stmt->get_result();

        if ($result->num_rows > 0) {
            $_SESSION['admin_error'] = 'Username sudah
digunakan!';
            header('Location: admin_user_add.php');
            exit();
        }

        // Cek apakah email sudah digunakan
        $stmt = $conn->prepare('SELECT id FROM users WHERE
email = ?');
        $stmt->bind_param('s', $email);
        $stmt->execute();
        $result = $stmt->get_result();

        if ($result->num_rows > 0) {
            $_SESSION['admin_error'] = 'Email sudah
digunakan!';
            header('Location: admin_user_add.php');
            exit();
        }

        // Hash password
        $hashed_password = password_hash($password,
PASSWORD_DEFAULT);
```

```

// Insert user baru
$stmt = $conn->prepare('INSERT INTO users
(username, email, password, role) VALUES (?, ?, ?, ?)');
$stmt->bind_param('ssss', $username, $email,
$hashed_password, $role);

if ($stmt->execute()) {
    $_SESSION['admin_success'] = 'User berhasil
ditambahkan!';
} else {
    $_SESSION['admin_error'] = 'Gagal menambahkan
user!';
}
} else {
    $_SESSION['admin_error'] = 'Data tidak lengkap!';
}
}

header('Location: admin_users.php');
exit();
?>

```

### **Listing program admin delet user**

```

<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';

$id = $_GET['id'] ?? '';
if ($id) {
    // Cek apakah user yang akan dihapus adalah admin yang
    sedang login
    if ($id == $_SESSION['admin_id']) {
        $_SESSION['admin_error'] = 'Tidak dapat menghapus
akun admin yang sedang login!';
    } else {
        // Cek apakah user ada
        $stmt = $conn->prepare('SELECT id FROM users WHERE
id = ?');
        $stmt->bind_param('i', $id);
        $stmt->execute();
        $result = $stmt->get_result();

        if ($result->num_rows > 0) {
            // Hapus user
            $stmt = $conn->prepare('DELETE FROM users
WHERE id = ?');
            $stmt->bind_param('i', $id);
        }
    }
}

```

```

        if ($stmt->execute()) {
            $_SESSION['admin_success'] = 'User
berhasil dihapus!';
        } else {
            $_SESSION['admin_error'] = 'Gagal
menghapus user!';
        }
    } else {
        $_SESSION['admin_error'] = 'User tidak
ditemukan!';
    }
} else {
    $_SESSION['admin_error'] = 'ID user tidak valid!';
}

header('Location: admin_users.php');
exit();
?>

```

### **Listing program admin user edit**

```

<?php
session_start();
if (!isset($_SESSION['admin_id'])) {
    header('Location: admin_login.php');
    exit();
}
require_once 'includes/db.php';

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $id = $_POST['id'] ?? '';
    $username = $_POST['username'] ?? '';
    $email = $_POST['email'] ?? '';
    $password = $_POST['password'] ?? '';
    $role = $_POST['role'] ?? 'user';

    if ($id && $username && $email && $role) {
        // Cek apakah username sudah digunakan oleh user
        $stmt = $conn->prepare('SELECT id FROM users WHERE
username = ? AND id != ?');
        $stmt->bind_param('si', $username, $id);
        $stmt->execute();
        $result = $stmt->get_result();

        if ($result->num_rows > 0) {
            $_SESSION['admin_error'] = 'Username sudah
digunakan!';
            header('Location: admin_user_edit.php?id=' .
$id);
        }
    }
}

```

```

        exit();
    }

    // Cek apakah email sudah digunakan oleh user lain
    $stmt = $conn->prepare('SELECT id FROM users WHERE
email = ? AND id != ?');
    $stmt->bind_param('si', $email, $id);
    $stmt->execute();
    $result = $stmt->get_result();

    if ($result->num_rows > 0) {
        $_SESSION['admin_error'] = 'Email sudah
digunakan!';
        header('Location: admin_user_edit.php?id=' .
$id);
        exit();
    }

    // Update user
    if (!empty($password)) {
        // Jika password diisi, update dengan password
baru
        $hashed_password = password_hash($password,
PASSWORD_DEFAULT);
        $stmt = $conn->prepare('UPDATE users SET
username=?, email=?, password=?, role=? WHERE id=?');
        $stmt->bind_param('ssssi', $username, $email,
$hashed_password, $role, $id);
    } else {
        // Jika password kosong, tidak update password
        $stmt = $conn->prepare('UPDATE users SET
username=?, email=?, role=? WHERE id=?');
        $stmt->bind_param('sssi', $username, $email,
$role, $id);
    }

    if ($stmt->execute()) {
        $_SESSION['admin_success'] = 'User berhasil
diupdate!';
    } else {
        $_SESSION['admin_error'] = 'Gagal mengupdate
user!';
    }
} else {
    $_SESSION['admin_error'] = 'Data tidak lengkap!';
}
}

header('Location: admin_users.php');
exit();
?>

```

### **Listing program logout admin**

```
<?php  
session_start();  
session_unset();  
session_destroy();  
header('Location: admin_login.php');  
exit();  
?>
```

## RIWAYAT HIDUP PENULIS



### Data diri

Nama	: Hasbi Kurnia Karim
Tempat Tgl/Lahir	: Bandung, 11 juli 2003
Jenis Kelamin	: Laki-Laki
Agama	: Islam
Status	: Belum Menikah
Nama Ayah	: Abdul Karim
Nama Ibu	: Heni Rosmiati

### Riwayat Pendidikan

- SDN Mandalahaji 2009-2015
- SMP As-Siroji 2015-2018
- MA Plus al- Barkah 2018-2021
- Universitas Bale Bandung (SI- Teknik Informatika) 2021-2025

### Motto

- Be grateful for yourself

### Contact

- No. telpon : 083818980574
- E-mail : karimhasbi@gmail.com

