

**APLIKASI E-LEARNING BERBASIS ANDROID PADA SMP  
NEGERI 2 PAREPARE**

**TUGAS AKHIR**

**Karya tulis sebagai salah satu syarat  
Untuk memperoleh gelar Sarjana dari  
Universitas Fajar**

Oleh

**FANDY FEBRIANSYAH**

**1720221039**



**PROGRAM STUDI TEKNIK ELEKTRO**

**FAKULTAS TEKNIK**

**UNIVERSITAS FAJAR**

**2023**

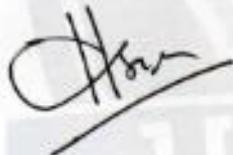
**HALAMAN PENGESAHAN**

**APLIKASI E-LEARNING BERBASIS ANDROID PADA SMP NEGERI 2  
PAREPARE**

Oleh  
**FANDY FEBRIANSYAH**  
1720221039

Menyetujui  
Tim Pembimbing  
Makassar, 22 Juni 2023

**Pembimbing I**



Asmawaty Azis, S.T., M.T.  
NIDN. 0905058504

**Pembimbing II**



Febriansyah, S.Kom., M.T.  
NIDN. 0921029003

Mengetahui,



**Dekan Fakultas Teknik**  
Prof. Dr. Erniati, S.T., M.T.  
NIDN. 0906107701

**Ketua Program Studi**



Safaruddin, S.Si., M.T.  
NIDN. 090910650

## PERNYATAAN ORISINALITAS

Penulis dengan ini menyatakan bahwa Tugas Akhir:

“Aplikasi E-Learning Berbasis Android pada SMP Negeri 2 Parepare” adalah karya orisinal saya dan setiap serta seluruh sumber acuan telah ditulis sesuai dengan Panduan Penulisan Ilmiah yang berlaku di Fakultas Teknik Universitas Fajar.

Makassar, 22 Juni 2023

Yang menyatakan



Fandy Febriansyah

## KATA PENGANTAR

Puji dan syukur penulis ucapkan kehadiran Tuhan Yang Maha Esa atas Berkat dan Karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir ini dengan judul “Aplikasi E-Learning Berbasis Android Pada SMP Negeri 2 Parepare”. Laporan penelitian ini menjadi salah satu syarat Program S1 Pendidikan Teknik Elektro Fakultas Teknik Universitas Fajar. Puji Syukur penulis dapat menyusun dengan baik walaupun masih ada kekurangan dalam pembuatannya.

Dalam penulisan laporan penelitian ini penulis mendapat banyak dukungan dan bantuan dari berbagai pihak sehingga penelitian ini dapat diselesaikan. Dengan ketulusan hati penulis menyampaikan ucapan terima kasih kepada :

1. Orang tua dan keluarga yang telah memberikan dukungan dan motivasi, serta pengorbanan materi dan doa dalam menyelesaikan tugas akhir ini.
2. Ibu Prof. Dr. Ir. Erniati, S.T.,M.T. selaku Dekan Fakultas Teknik Universitas Fajar.
3. Ibu Kurniawan Harun Rasyid, S.T.,M.T. selaku penasehat akademik yang telah membimbing dalam segala proses.
4. Bapak Safaruddin, S.Si., MT. selaku Ketua Program Studi Teknik Elektro Universitas Fajar
5. Ibu Asmawaty Azis, S.T.,M.T. selaku Dosen Pembimbing I.
6. Bapak Febriansyah, S.Kom.,M.T. selaku Dosen Pembimbing II
7. Dosen-dosen Prodi Teknik Elektro Universitas Fajar
8. Teman-teman Program Studi Teknik Elektro 2017 untuk kebersamaannya selama ini serta masukan dan arahan dalam penulisan proposal skripsi.
9. HME FT-UNIFA yang telah memberikan kami kesempatan berlembaga.
10. KBMFT-UNIFA yang telah menerima kami menjadi keluarga.
11. Seluruh orang-orang yang telah membantu penulis selama proses pengerjaan tugas akhir ini.

Semoga Tuhan Yang Maha Esa senantiasa melimpahkan berkat dan karunia-Nya kepada kami atas usaha, perjuangan dan pengorbanan yang dilakukan.

Akhir kata penulis berharap laporan penelitian ini dapat membantu para pembaca dan pihak-pihak yang berkaitan dengan penelitian ini

Makassar, 20 Februari 2023

Fandy Febriansyah

## ABSTRAK

**Aplikasi E-Learning Berbasis Android Pada SMP Negeri 2 Parepare, Fandy Febriansyah.** SMP Negeri 2 Parepare adalah sekolah yang terletak di Kota Parepare yang dalam proses belajar mengajar antara guru dan siswa masih secara konvensional, dimana proses pembelajaran ini dilakukan secara tatap muka di dalam kelas dan wilayah sekolah pada jam pelajaran yang telah ditetapkan. Terbatasnya waktu belajar-mengajar yang membuat guru tidak bisa memberikan tugas dan kuis untuk siswa, materi yang tidak bisa disampaikan jika banyaknya materi yang akan diajarkan serta ketika guru berhalangan hadir dikelas membuat proses belajar mengajar kurang optimal. Penelitian ini bertujuan untuk menghasilkan sistem *e-learning* berbasis Android pada SMP Negeri 2 Parepare, menghubungkan data pada *web* dan aplikasi Android, serta mengetahui efektivitas penerapan *e-learning* terhadap proses pembelajaran berbasis Android. Metode yang digunakan dalam penelitian ini adalah merancang aplikasi berbasis Android pada SMP Negeri 2 Parepare menggunakan Laravel dan Flutter. Hasil dari pembuatan aplikasi *e-learning* ini dapat membantu proses pembelajaran dalam mengakomodasi masalah pemberian materi pelajaran, pengerjaan tugas dan kuis yang tidak harus selalu di kelas, dan melakukan ujian *online*. Melalui hasil pengujian kuesioner dari 20 responden menyatakan “sangat setuju” bahwa aplikasi *e-learning* berbasis android ini layak dan efektif terhadap proses pembelajaran dengan perolehan nilai 88,8%.

**Kata kunci:** Aplikasi, *E-Learning*, Android, SMP Negeri 2 Parepare, Laravel, Flutter.

## ABSTRACT

**Android Based E-Learning Application at SMP Negeri 2 Parepare, Fandy Febriansyah.** SMP SMP Negeri 2 Parepare is a school located in the city of Parepare where the teaching and learning process between teachers and students is still conventional, where the learning process is carried out face to face in the classroom and school area during predetermined lesson hours. Limited teaching and learning time makes the teacher unable to give assignments and quizzes for students, material that cannot be delivered if there is a lot of material to be taught and when the teacher is unable to attend class makes the teaching and learning process less than optimal. This study aims to produce an Android-based e-learning system at Parepare 2 Public Middle School, connect data on the web and Android applications, and determine the effectiveness of implementing e-learning on Android-based learning processes. The method used in this research is to design an Android-based application at SMP Negeri 2 Parepare using Laravel and Flutter. The results of making this e-learning application can help the learning process in accommodating problems in providing subject matter, doing assignments and quizzes that don't have to be in class, and conducting online exams. Through the results of testing the questionnaire, 20 respondents stated "strongly agree" that this Android-based e-learning application is feasible and effective in the learning process with an acquisition value of 88.8%.

**Keywords:** Application, E-Learning, Android, SMP Negeri 2 Parepare, Laravel, Flutter.

## DAFTAR ISI

<b>SAMPUL</b> .....	i
<b>HALAMAN PENGESAHAN</b> .....	Error! Bookmark not defined.
<b>PERNYATAAN ORISINALITAS</b> .....	Error! Bookmark not defined.
<b>KATA PENGANTAR</b> .....	iv
<b>ABSTRAK</b> .....	vi
<b>ABSTRACT</b> .....	vii
<b>DAFTAR ISI</b> .....	viii
<b>DAFTAR TABEL</b> .....	xi
<b>DAFTAR GAMBAR</b> .....	xiii
<b>BAB I PENDAHULUAN</b> .....	1
I.1    Latar Belakang.....	1
I.2    Rumusan Masalah .....	3
I.3    Tujuan Penelitian.....	3
I.4    Batasan Masalah.....	3
<b>BAB II TINJAUAN PUSTAKA</b> .....	4
II.1    Kerangka Teori.....	4
II.1.1 <i>E-Learning</i> .....	4
II.1.2    Komponen <i>E-Learning</i> .....	6
II.1.3    Aplikasi .....	6
II.1.4    Android .....	7
II.1.5    UML ( <i>Unified Modelling Language</i> ).....	8
II.1.6    Android Studio .....	15
II.1.7 <i>Website</i> .....	15
II.1.8    Bahasa Pemrograman.....	16
II.1.9    PHP .....	16
II.1.10 <i>Framework</i> Laravel.....	18
II.1.11    Flutter .....	19

II.1.12	Visual Studio Code .....	20
II.1.13	MySQL.....	21
II.1.14	MVC ( <i>Model View Controller</i> ).....	21
II.1.15	API ( <i>Application Programming Interface</i> ).....	23
II.1.16	Pengujian.....	24
II.2	Penelitian Terdahulu.....	25
II.3	Kerangka Pemikiran .....	33
<b>BAB III</b>	<b>METODE PENELITIAN .....</b>	<b>35</b>
III.1	Bagan Alur Penelitian.....	35
III.2	Metode Pengembangan Sistem.....	36
III.3	Rancangan Penelitian .....	38
III.2.1	Sistem yang Berjalan.....	38
III.2.2	Sistem yang Diusulkan.....	39
III.4	Waktu dan Lokasi Penelitian.....	41
III.5	Alat dan Bahan .....	42
III.6	Metode Pengumpulan Data .....	42
III.7	Analisis Sistem .....	43
III.6.1	<i>Use Case Diagram</i> .....	43
III.6.2	<i>Activity Diagram</i> .....	44
III.6.3	<i>Sequence Diagram</i> .....	56
III.6.4	<i>Class Diagram</i> .....	58
III.6.5	<i>User Interface</i> .....	59
<b>BAB IV</b>	<b>HASIL DAN PEMBAHASAN.....</b>	<b>62</b>
IV.1	Hasil.....	62
IV.1.1	Tampilan Admin .....	62
IV.1.2	Tampilan Guru .....	65
IV.1.3	Tampilan Siswa.....	68
IV.2	Pembahasan .....	76
IV.2.1	Pengujian <i>Black Box</i> .....	76

IV.2.2	Pengujian <i>White Box</i> .....	81
IV.2.3	Pengujian Efektivitas Penerapan Sistem <i>E-Learning</i> .....	111
IV.2.4	Implementasi API <i>Website</i> pada Aplikasi Android .....	117
<b>BAB V</b>	<b>PENUTUP</b> .....	<b>123</b>
V.1	Kesimpulan.....	123
V.2	Saran.....	123
<b>DAFTAR PUSTAKA</b>	.....	<b>124</b>
<b>LAMPIRAN</b>	.....	<b>128</b>

## DAFTAR TABEL

Tabel II.1	Simbol <i>Use Case Diagram</i> .....	8
Tabel II.2	Simbol <i>Activity Diagram</i> .....	11
Tabel II.3	Simbol <i>Sequence Diagram</i> .....	12
Tabel II.4	Simbol <i>Class Diagram</i> .....	14
Tabel II.5	<i>State Of The Art</i> .....	25
Tabel III.1	Perangkat Keras .....	42
Tabel IV.1	Instrumen Pengujian <i>Black Box</i> pada <i>Website Admin</i> .....	76
Tabel IV.2	Instrumen Pengujian <i>Black Box</i> pada <i>Website Guru</i> .....	78
Tabel IV.3	Instrumen Pengujian <i>Black Box</i> pada <i>Android Siswa</i> .....	80
Tabel IV.4	<i>Test Case Login</i> .....	82
Tabel IV.5	<i>Test Case Data Kelas</i> oleh Admin .....	84
Tabel IV.6	<i>Test Case Data Siswa</i> oleh Admin .....	87
Tabel IV.7	<i>Test Case Data Guru</i> oleh Admin .....	89
Tabel IV.8	<i>Test Case Data Mapel</i> oleh Admin .....	91
Tabel IV.9	<i>Test Case Data Jadwal</i> oleh Admin .....	93
Tabel IV.10	<i>Test Case Data Materi</i> oleh Guru .....	96
Tabel IV.11	<i>Test Case Data Tugas</i> oleh Guru .....	98
Tabel IV.12	<i>Test Case Data Kuis</i> oleh Guru .....	100
Tabel IV.13	<i>Test Case Data Ujian</i> oleh Guru .....	102
Tabel IV.14	<i>Test Case Akses Materi</i> oleh Siswa .....	105
Tabel IV.15	<i>Test Case Akses Tugas</i> oleh Siswa .....	106
Tabel IV.16	<i>Test Case Akses Kuis</i> oleh Siswa .....	108
Tabel IV.17	<i>Test Case Akses Ujian</i> oleh Siswa .....	110
Tabel IV.18	Nilai Skor Jawaban .....	111
Tabel IV.19	Kriteria Skor Berdasarkan Interval .....	112
Tabel IV.20	Hasil Perhitungan Pertanyaan 1 .....	112
Tabel IV.21	Hasil Perhitungan Pertanyaan 2 .....	112
Tabel IV.22	Hasil Perhitungan Pertanyaan 3 .....	113
Tabel IV.23	Hasil Perhitungan Pertanyaan 4 .....	113

Tabel IV.24	Hasil Perhitungan Pertanyaan 5.....	114
Tabel IV.25	Hasil Perhitungan Pertanyaan 6.....	114
Tabel IV.26	Hasil Perhitungan Pertanyaan 7.....	115
Tabel IV.27	Hasil Perhitungan Pertanyaan 8.....	115
Tabel IV.28	Hasil Perhitungan Pertanyaan 9.....	116
Tabel IV.29	Hasil Perhitungan Pertanyaan 10.....	116

## DAFTAR GAMBAR

Gambar II.1	Bahasa Pemrograman PHP .....	17
Gambar II.2	<i>Framework</i> Laravel.....	18
Gambar II.3	Flutter.....	19
Gambar II.4	Visual Studio Code .....	20
Gambar II.5	MySQL .....	21
Gambar II.6	Skema MVC .....	22
Gambar II.7	Analogi API pada Pembangunan Rumah .....	23
Gambar II.8	Skema Konektivitas API Antar <i>Software</i> .....	24
Gambar II.9	Kerangka Pemikiran .....	34
Gambar III.1	Diagram Alur Penelitian .....	35
Gambar III.2	Metode <i>Waterfall</i> .....	37
Gambar III.3	Sistem yang Berjalan .....	39
Gambar III.4	Sistem Pembelajaran yang Diusulkan.....	39
Gambar III.5	Sistem Pemberian-Pengerjaan Tugas yang Diusulkan .....	40
Gambar III.6	Sistem Pemberian-Pengerjaan Kuis yang Diusulkan.....	40
Gambar III.7	Sistem Pelaksanaan Ujian yang Diusulkan.....	41
Gambar III.8	<i>Use Case Diagram</i> .....	43
Gambar III.9	<i>Activity Diagram Login</i> .....	44
Gambar III.10	<i>Activity Diagram</i> Kelola Data Kelas .....	45
Gambar III.11	<i>Activity Diagram</i> Kelola Data Siswa.....	46
Gambar III.12	<i>Activity Diagram</i> Kelola Data Guru .....	47
Gambar III.13	<i>Activity Diagram</i> Kelola Data Mapel .....	48
Gambar III.14	<i>Activity Diagram</i> Kelola Data Jadwal .....	49
Gambar III.15	<i>Activity Diagram</i> Kelola Data Materi.....	50
Gambar III.16	<i>Activity Diagram</i> Kelola Data Tugas.....	51
Gambar III.17	<i>Activity Diagram</i> Kelola Data Kuis .....	52
Gambar III.18	<i>Activity Diagram</i> Kelola Data Ujian.....	53
Gambar III.19	<i>Activity Diagram</i> Akses Data Materi.....	54
Gambar III.20	<i>Activity Diagram</i> Akses Data Tugas.....	54

Gambar III.21	<i>Activity Diagram</i> Akses Data Kuis .....	55
Gambar III.22	<i>Activity Diagram</i> Akses Data Ujian.....	55
Gambar III.23	<i>Sequence Diagram</i> Login.....	56
Gambar III.24	<i>Sequence Diagram</i> Kelola <i>E-Learning</i> Admin.....	56
Gambar III.25	<i>Sequence Diagram</i> Kelola <i>E-Learning</i> Guru.....	57
Gambar III.26	<i>Sequence Diagram</i> Akses <i>E-Learning</i> Siswa .....	58
Gambar III.27	<i>Class Diagram</i> <i>E-Learning</i> .....	58
Gambar III.28	<i>Interface</i> Login Admin dan Guru.....	59
Gambar III.29	<i>Interface</i> Login Siswa .....	59
Gambar III.30	<i>Interface</i> Web Admin.....	60
Gambar III.31	<i>Interface</i> Web Guru.....	60
Gambar III.32	<i>Interface</i> Aplikasi Android Siswa.....	61
Gambar IV.1	Tampilan <i>Login</i> Admin.....	62
Gambar IV.2	Tampilan <i>Dashboard</i> Admin .....	62
Gambar IV.3	Tampilan Data Kelas .....	63
Gambar IV.4	Tampilan Data Siswa .....	63
Gambar IV.5	Tampilan Data Guru .....	64
Gambar IV.6	Tampilan Data Mapel .....	64
Gambar IV.7	Tampilan Data Jadwal .....	65
Gambar IV.8	Tampilan <i>Dashboard</i> Guru .....	65
Gambar IV.9	Tampilan Kelola Materi.....	66
Gambar IV.10	Tampilan Kelola Tugas.....	66
Gambar IV.11	Tampilan Kelola Kuis.....	67
Gambar IV.12	Tampilan Kelola Ujian.....	67
Gambar IV.13	Tampilan <i>Login</i> Siswa .....	68
Gambar IV.14	Tampilan Menu Siswa .....	68
Gambar IV.15	Tampilan <i>Dashboard</i> Siswa.....	69
Gambar IV.16	Tampilan Jadwal Siswa .....	69
Gambar IV.17	Tampilan Daftar Mata Pelajaran.....	70
Gambar IV.18	Tampilan Materi Siswa.....	70
Gambar IV.19	Tampilan Detail Materi.....	71

Gambar IV.20	Tampilan Tugas Siswa.....	71
Gambar IV.21	Tampilan <i>Submit</i> Tugas .....	72
Gambar IV.22	Tampilan Kuis Siswa .....	72
Gambar IV.23	Tampilan <i>Submit</i> Kuis .....	73
Gambar IV.24	Tampilan Ujian Siswa.....	73
Gambar IV.25	Tampilan <i>Scan QR</i> .....	74
Gambar IV.26	Tampilan Soal Ujian .....	75
Gambar IV.27	Pengacakan Soal Pilihan Ganda .....	75
Gambar IV.28	<i>Flowchart Login</i> .....	81
Gambar IV.29	<i>Flowgraph Login</i> .....	82
Gambar IV.30	<i>Flowchart</i> Data Kelas oleh Admin .....	83
Gambar IV.31	<i>Flowgraph</i> Data Kelas oleh Admin .....	84
Gambar IV.32	<i>Flowchart</i> Data Siswa oleh Admin.....	86
Gambar IV.33	<i>Flowgraph</i> Data Siswa oleh Admin.....	86
Gambar IV.34	<i>Flowchart</i> Data Guru oleh Admin .....	88
Gambar IV.35	<i>Flowgraph</i> Data Guru oleh Admin.....	88
Gambar IV.36	<i>Flowchart</i> Data Mapel oleh Admin.....	90
Gambar IV.37	<i>Flowgraph</i> Data Mapel oleh Admin.....	90
Gambar IV.38	<i>Flowchart</i> Data Jadwal oleh Admin .....	92
Gambar IV.39	<i>Flowgraph</i> Data Jadwal oleh Admin .....	93
Gambar IV.40	<i>Flowchart</i> Data Materi oleh Guru .....	95
Gambar IV.41	<i>Flowgraph</i> Data Materi oleh Guru .....	95
Gambar IV.42	<i>Flowchart</i> Data Tugas oleh Guru .....	97
Gambar IV.43	<i>Flowgraph</i> Data Tugas oleh Guru .....	97
Gambar IV.44	<i>Flowchart</i> Data Kuis oleh Guru.....	99
Gambar IV.45	<i>Flowgraph</i> Data Kuis oleh Guru .....	99
Gambar IV.46	<i>Flowchart</i> Data Ujian oleh Guru .....	101
Gambar IV.47	<i>Flowgraph</i> Data Ujian oleh Guru .....	102
Gambar IV.48	<i>Flowchart</i> Akses Materi oleh Siswa.....	104
Gambar IV.49	<i>Flowgraph</i> Akses Materi oleh Siswa.....	104
Gambar IV.50	<i>Flowchart</i> Akses Tugas oleh Siswa.....	105

Gambar IV.51	<i>Flowgraph</i> Akses Tugas oleh Siswa.....	105
Gambar IV.52	<i>Flowchart</i> Akses Kuis oleh Siswa .....	107
Gambar IV.53	<i>Flowgraph</i> Akses Kuis oleh Siswa.....	107
Gambar IV.54	<i>Flowchart</i> Akses Ujian oleh Siswa.....	109
Gambar IV.55	<i>Flowgraph</i> Akses Ujian oleh Siswa.....	109

# BAB I PENDAHULUAN

## I.1 Latar Belakang

Pembelajaran *e-learning* berbasis Android saat ini menjadi salah satu dari sekian bentuk pembelajaran yang dikembangkan dan diimplementasikan sebagai alat tambahan untuk menunjang kegiatan pembelajaran. *E-learning* adalah pembelajaran jarak jauh yang menggunakan media elektronik untuk menyampaikan pembelajaran, seperti *smartphone*, dan didukung oleh koneksi internet. Pembelajaran *e-learning* diciptakan hanya sebagai lingkungan belajar alternatif, bukan untuk menggantikan sistem belajar mengajar yang ada. Pembelajaran tatap muka langsung dan pembelajaran *online* melalui *e-learning* akan digabungkan untuk membangun sistem pembelajaran *e-learning* campuran yang memadukan sistem pembelajaran tatap muka tradisional dengan sistem pembelajaran kelas *virtual e-learning*. Pembelajaran *online* dapat berlangsung di beberapa lokasi dan pada waktu yang lebih fleksibel karena dapat dilakukan tanpa kehadiran siswa atau guru di kelas.

SMP Negeri 2 Parepare merupakan sekolah menengah pertama negeri yang ada di Kota Parepare, yang terletak di Jl. Lahalede No.84, Kelurahan Lakessi, Kecamatan Soreang, Sulawesi Selatan. Sekolah Menengah Pertama Negeri (SMPN) 2 Rujukan Kota Parepare dengan visi Mewujudkan siswa SMP NEGERI 2 PAREPARE yang “Unggul dalam prestasi, Kompetitif, *Religius*, Berkarakter dan Berwawasan lingkungan”, selalu mengapresiasi bakat dan minat setiap peserta didik untuk berkembang, baik di bidang akademik maupun non akademik. Permasalahan yang muncul di SMP Negeri 2 Parepare, salah satunya adalah belum banyak interaksi antara guru dan siswa dalam cara belajar mengajar. Hanya satu atau dua jam untuk satu mapel. Kurang interaksi juga ada antara guru dan siswa. Oleh karena itu jika ada kesulitan, seperti ketika siswa merasa sulit untuk mengulang materi karena mereka tidak memahami mata pelajaran tertentu, mereka harus bertanya langsung kepada guru yang

bersangkutan. Oleh karena itu, dibutuhkan aplikasi yang dapat mendukung proses belajar mengajar antara guru dan siswa setiap saat dan dari mana saja, memungkinkan siswa untuk memahami materi pelajaran secara lebih menyeluruh, meningkatkan minat dan prestasi siswa, serta mengintensifkan interaksi antara guru dan siswa di luar kelas.

Terdapat beberapa penelitian yang cukup relevan mengenai aplikasi yang dijadikan sebagai referensi dalam melakukan penelitian yaitu, penelitian yang dilakukan oleh Susanto dkk (2017) “Perancangan *E-Learning* Berbasis Web Pada SMP Negeri 3 Patuk Gunungkidul Yogyakarta”, hasil dari penelitian ini Temuan penelitian ini menunjukkan bahwa e-learning dapat digunakan sebagai alat untuk meningkatkan proses belajar mengajar antara guru dan siswa. Alat ini dapat diakses dari lokasi mana saja kapan saja dan dapat meningkatkan tingkat kontak antara guru dan siswa. Penelitian selanjutnya yang dilakukan oleh Hoetamy dkk (2020) “Perancangan *E-Learning* Pada SMK Veteran Kota Jambi Berbasis Android”, hasil dari penelitian ini adalah *e-learning* yang dibangun dapat digunakan untuk membuat soal latihan dan mengunggah materi dan video pembelajaran, *e-learning* ini dikembangkan menggunakan bahasa pemrograman Java dan *database* MySQL.

Keberadaan “Aplikasi *E-Learning* Berbasis Android pada SMP Negeri 2 Parepare” ini, diharapkan dapat membantu permasalahan yang terjadi. Media pembelajaran *online* ini dapat membantu proses belajar mengajar dengan memungkinkan guru untuk terus memberikan pengetahuan kepada siswa bahkan ketika mereka tidak hadir secara fisik di sekolah. Karena guru dapat memanfaatkan sumber belajar *online* untuk memberikan materi, tugas, kuis serta memberikan ujian kepada para siswa tanpa harus menghadiri kelas secara fisik. Di sisi lain, siswa dapat menerima apa yang dikatakan oleh guru meskipun ketika mereka tidak berinteraksi langsung dengan guru dan ketika mereka tidak dapat menghadiri kelas. (Susanto dkk, 2016).

## **I.2 Rumusan Masalah**

Adapun rumusan masalah yang diharapkan dapat diselesaikan dengan penelitian ini adalah sebagai berikut:

1. Bagaimana membuat sebuah aplikasi *e-learning* berbasis Android untuk SMP Negeri 2 Parepare?
2. Bagaimana menghubungkan data pada *web* dan Android?
3. Bagaimana efektivitas penerapan *e-learning* terhadap proses pembelajaran berbasis Android?

## **I.3 Tujuan Penelitian**

Adapun tujuan-tujuan yang diharapkan dapat diselesaikan adalah sebagai berikut:

1. Membuat sebuah aplikasi *e-learning* berbasis Android untuk SMP Negeri 2 Parepare.
2. Menghubungkan data pada *web* dan Android menggunakan API yang tercipta dari aplikasi *web* yang diimplementasikan dengan aplikasi Android.
3. Untuk mengetahui efektivitas penerapan *e-learning* terhadap proses pembelajaran berbasis Android.

## **I.4 Batasan Masalah**

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Aplikasi yang dibuat tidak bisa digunakan tanpa koneksi internet.
2. Aplikasi yang dibuat tidak memiliki fitur untuk komunikasi langsung antara guru dan siswa.

## BAB II TINJAUAN PUSTAKA

### II.1 Kerangka Teori

#### II.1.1 *E-Learning*

Pendekatan baru untuk pengajaran dan pembelajaran menggunakan media elektronik, khususnya *internet* sebagai sistem pembelajaran, dikenal sebagai "*e-learning*" (Khamidah & Triyono, 2013). Dalam lingkungan belajar yang terbuka, fleksibel, dan tersebar, *e-learning* mengacu pada penyampaian informasi pendidikan kepada siapa saja, di mana saja, dan kapan saja. Selain itu, istilah "pembelajaran terbuka dan fleksibel" mengacu pada kebebasan peserta didik sehubungan dengan waktu, tempat, kecepatan, konten materi, gaya belajar, jenis penilaian, dan pembelajaran mandiri atau kolaboratif. (Khan dalam Suartama, 2014).

Banyaknya jenis sistem *e-learning* yang tersedia, secara umum dapat dikategorikan tergantung pada tingkat interaksinya dan dipisahkan menjadi dua kelompok, yaitu *e-learning* statis dan *e-learning* dinamis. 1) Ketika peserta didik hanya dapat mengunduh materi yang diperlukan, administrator hanya dapat mengunggah file materi, dan pengguna sistem tidak dapat berkomunikasi satu sama lain, sistem *e-learning* dianggap statis. Biasanya, sistem ini hanya digunakan untuk membantu kegiatan belajar mengajar di kelas secara langsung. 2) Sementara itu sistem *e-learning* dianggap dinamis ketika siswa dapat belajar dalam suasana yang tidak berbeda secara signifikan dengan lingkungan kelas dan dimana terdapat potensi interaksi antara siswa dan guru melalui *e-mail*, forum diskusi, *chatting*, atau sarana komunikasi lainnya (Suteja & Harjoko dalam Khamidah & Triyono, 2013).

Menurut Siahaan dalam Hadisi dan Muna (2015) untuk kegiatan pembelajaran di kelas (*Classroom Instruction*), *e-learning* setidaknya memiliki tiga fungsi. Diantaranya sebagai berikut:

1. Tambahan (*Supplement*)

Ketika siswa memiliki pilihan untuk menggunakan atau tidak menggunakan sumber belajar elektronik, itu dianggap berfungsi sebagai "tambahan". Peserta didik tidak diharuskan dalam situasi ini untuk menggunakan sumber belajar elektronik. Meskipun bersifat opsional, siswa yang menggunakannya niscaya akan mendapatkan informasi atau wawasan baru.

2. Pelengkap (*Complement*)

Jika alat pembelajaran elektronik dirancang untuk menemani materi pembelajaran yang siswa dapatkan di kelas, mereka disebut "pelengkap" materi tersebut. Sumber belajar elektronik dirancang untuk melengkapi kegiatan pembelajaran tradisional dengan berfungsi sebagai penguatan atau materi perbaikan bagi siswa.

3. Pengganti (*Substitution*)

Banyak model alternatif yang berbeda dari kegiatan pendidikan dan kuliah yang ditawarkan kepada peserta didik oleh beberapa sekolah atau perguruan tinggi di negara-negara maju. Tujuannya agar mahasiswa mampu menangani kegiatan perkuliahannya secara fleksibel sesuai dengan jadwal harian dan kewajiban lainnya. Berikut 3 alternatif model kegiatan pembelajaran yang dapat dipilih:

- a. Sepenuhnya secara tatap muka (konvensional)
- b. Sebagian secara tatap muka dan sebagian lagi melalui internet, atau bahkan
- c. Sepenuhnya melalui internet.

Model pembelajaran alternatif yang dipilih peserta didik tidak ada hubungannya dengan evaluasi. Karena ketiga metode diakui atau dievaluasi secara setara. Lembaga pendidikan akan mengenali peserta didik secara merata apakah mereka menyelesaikan pendidikan mereka dan lulus melalui metode tradisional, benar-benar online, atau bahkan hibrida dari kedua metode sebelumnya. Kondisi yang sangat fleksibel ini

dipandang cukup bermanfaat bagi peserta didik yang ingin menyelesaikan pendidikan mereka lebih cepat.

### **II.1.2 Komponen *E-Learning***

Jika didukung oleh bagian-bagian penyusunnya, pembelajaran *e-learning* dapat berhasil diselesaikan. *E-learning* terdiri dari elemen-elemen berikut:

#### 1. Infrastruktur *e-learning*

Berupa jaringan komputer, *personal computer* (pc), internet dan perlengkapan multimedia, hingga *smartphone* untuk mendukung jalannya aplikasi *e-learning*.

#### 2. Sistem dan aplikasi *e-learning*

Berupa perangkat lunak yang dapat membantu pembentukan *virtual* dari proses belajar mengajar konvensional seperti manajemen kelas, pembuatan materi atau konten, sistem evaluasi, sistem ujian *online*. Sistem serta aplikasi *e-learning* ini biasa disebut dengan *Learning Management System* (LMS).

#### 3. Konten *e-learning*

Materi pembelajaran *e-learning* hadir dalam bentuk materi pembelajaran berbasis teks dan multimedia. Kedua jenis konten tersebut dapat disimpan di LMS sehingga siswa dapat mengaksesnya kapan pun dan di mana pun mereka berada.

### **II.1.3 Aplikasi**

Menurut Sibarani dkk (2017) Istilah "aplikasi" berasal dari kata bahasa Inggris "*application*", yang juga menunjukkan penggunaan atau penerapan. Sementara program siap pakai yang dimaksudkan untuk melakukan tugas bagi pengguna layanan aplikasi dan penggunaan aplikasi lain yang dapat digunakan oleh target yang dituju adalah definisi aplikasi, itu juga merupakan program siap pakai. Aplikasi juga mengacu pada praktik penerapan salah satu pendekatan pemrosesan data aplikasi untuk memecahkan masalah yang biasanya bersaing dengan perhitungan yang diinginkan atau diantisipasi atau pemrosesan data

yang diantisipasi. (Juansyah, 2015). Berdasarkan interpretasi tersebut, aplikasi adalah bagian dari perangkat lunak komputer yang dapat mengatasi masalah tertentu sesuai dengan kebutuhan sistem.

#### **II.1.4 Android**

Menurut Juansyah (2015) menyatakan bahwa sistem operasi, *middleware*, dan aplikasi semuanya termasuk dalam sistem operasi perangkat seluler berbasis Linux *open source* yang dikenal sebagai Android. Platform terbuka disediakan oleh Android untuk pengembangan aplikasi. Untuk *smartphone* dan *tablet*, ada sistem operasi yang disebut Android. Sistem operasi dapat dianggap sebagai "jembatan" yang menghubungkan perangkat dan penggunanya, memungkinkan interaksi dan penggunaan aplikasi yang diinstal pada perangkat. (Kuswanto, 2018).

Menurut Hamka dkk. (2015) Android memiliki sejumlah keunggulan yang mempercepat pengembangan sistem operasi ini, di antaranya:

1. Karena sistem operasi ini adalah *open source*, siapa saja dapat dengan bebas mengembangkannya tanpa harus membayar biaya lisensi apa pun kepada Android.
2. Fitur *Global Positioning System* (GPS), penyimpanan data lokal menggunakan SQL, dan masih banyak lagi fitur dan fungsi yang ada di antara fitur atau fungsi yang komprehensif dan beragam yang telah dihasilkan.
3. Dengan *antialiased* 2D dan 3D *Accelerated OpenGL* dan *codec* untuk mendukung AVC, MP3, AAC, OGG, dan banyak format audio lainnya, visual sistem operasi ini cukup bagus.

Sistem operasi Android digunakan oleh sejumlah perangkat yang berbeda, termasuk TV, kamera, dan lainnya, selain perangkat seluler (*smartphone*).

### II.1.5 UML (*Unified Modelling Language*)

Menurut Sukamto dan Shalahuddin (2018:133), mendefinisikan bahwa UML adalah bahasa standar yang digunakan dalam pemrograman berorientasi objek untuk menganalisis, membuat, dan menggambarkan arsitektur program.

UML (*Unified Modeling Language*) adalah pengganti metode analisis dan desain berorientasi objek (OOAD&D/*object oriented analysis and design*) yang muncul pada akhir 1980-an dan awal 1990-an. Menggabungkan pendekatan Booch, Rumbaugh (OMT), dan Jacobson menghasilkan UML. Namun, UML lebih komprehensif daripada OOAD. Dengan tujuan UML (yang sudah banyak digunakan oleh berbagai kalangan) menjadi bahasa pemodelan standar di masa depan, standardisasi proses dilakukan dengan OMG (*Object Management Group*) sepanjang penciptaan UML. (Pratama dalam Kurniawan, 2020).

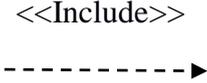
Karena UML sangat komprehensif dan terperinci, UML juga dapat berfungsi sebagai cetak biru. Dengan bantuan cetak biru ini, dimungkinkan untuk mempelajari semua tentang pengkodean program atau bahkan membaca kode dan menafsirkannya kembali menggunakan diagram (*reverse engineering*) (Wati dan Kusumo, 2016). Jenis diagram meliputi :

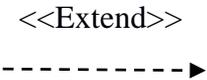
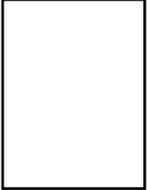
#### 1. *Use Case Diagram*

*Use case diagram* menunjukkan bagaimana sistem seharusnya bekerja. Sistem "apa" daripada "bagaimana" disorot. Pada sebuah *use case* menggambarkan sebuah tugas tertentu, seperti masuk ke sistem lalu membuat daftar belanja, dan lainnya. Aktor adalah entitas manusia atau mekanik yang berinteraksi dengan suatu sistem untuk melaksanakan tujuan tertentu. (Dharwiyanti dan Wahono, 2003).

**Tabel II.1** Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1		<i>Use Case</i>	Pada awal frase nama <i>use case</i> sering dimulai dengan kata kerja untuk menunjukkan fungsionalitas yang ditawarkan sistem sebagai unit yang

			berkomunikasi dengan unit atau aktor lain.
2		<i>Actor</i>	Mencerminkan fungsi entitas di luar sistem informasi yang akan dibangun sendiri (orang, proses, atau sistem lainnya) yang berinteraksi dengan sistem informasi yang akan dibuat. Meskipun simbol aktor adalah gambar seseorang, aktor tidak selalu seseorang. Ini biasanya dinyatakan dengan menggunakan kata benda di awal frasa nama aktor.
3		<i>Association</i>	Keterlibatan aktor dalam <i>use case</i> ditunjukkan oleh simbol garis yang menghubungkan aktor dengan <i>use case</i> .
4		<i>Generalization</i>	Hubungan antara dua <i>use case</i> di mana satu fungsi lebih umum daripada yang lain dalam hal generalisasi dan spesialisasi (generik-khusus).
5		<i>Include</i>	Hubungan antara dua <i>use case</i> di mana <i>use case</i> yang dijalankan diperlukan agar kasus penggunaan tambahan berfungsi atau sebagai prasyarat.

6		<i>Extend</i>	<p>Mirip dengan konsep pewarisan dalam pemrograman berorientasi objek, <i>use case</i> baru terkait dengan <i>use case</i> yang ada dengan cara yang memungkinkan <i>use case</i> tambahan berfungsi secara independen bahkan tanpa <i>use case</i> tambahan. Biasanya, <i>use case</i> tambahan memiliki nama awal yang sama dengan <i>use case</i>.</p>
7		<i>Dependency</i>	<p>Perubahan dalam satu elemen independen akan berdampak pada elemen yang bergantung pada elemen independen tersebut.</p>
8		<i>System</i>	<p>Entitas yang menanggapi permintaan aktor dan menawarkan layanan yang diperlukan dilambangkan dengan kotak.</p>
9		<i>Collaboration</i>	<p>Interaksi hukum dan komponen lain yang bergabung untuk meningkatkan perilaku agregat bagian-bagian (sinergi).</p>

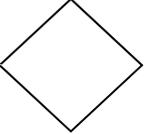
10		<i>Note</i>	Ketika sebuah program dijalankan, komponen fisik yang mewakili sumber daya komputer ada di sana.
----	---	-------------	--

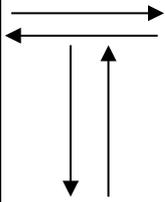
Sumber: Rosa A.S dan M. Shalahuddin, 2013

## 2. Activity Diagram

*Activity diagram* menunjukkan banyak aliran aktivitas dalam sistem yang akan dibuat, bagaimana setiap aliran dimulai, keputusan apa yang dapat diambil di sepanjang jalan, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menunjukkan proses bersamaan yang mungkin terjadi dalam proses yang berbeda. (Dharwiyanti dan Wahono, 2003).

**Tabel II.2** Simbol *Activity Diagram*

No	Gambar	Nama	Keterangan
1		<i>Activity</i>	<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis.
2		<i>Decision</i>	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i>
3		<i>Start</i>	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas
4		<i>End</i>	<i>End Point</i> , akhir aktivitas
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

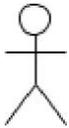
7		<i>Swimlane</i>	<i>Swimlane, pembagian activity diagram untuk menunjukkan siapa melakukan apa</i>
8		<i>Line Connector</i>	<i>Digunakan untuk menghubungkan satu simbol dengan simbol lainnya.</i>

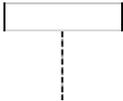
Sumber: Rosa A.S dan M. Shalahuddin, 2013

### 3. *Sequence Diagram*

Dalam hal pesan yang ditampilkan terhadap waktu, *sequence diagram* menunjukkan interaksi antara objek sistem (seperti pengguna, tampilan, dll.). Dimensi horizontal (item terkait) dan dimensi vertikal (waktu) membentuk *sequence diagram*. Skenario atau serangkaian tindakan yang dilakukan sebagai reaksi terhadap suatu peristiwa untuk mencapai hasil tertentu sering diilustrasikan menggunakan *sequence diagram*. Mengapa kegiatan ini dimulai, proses dan perubahan internal apa yang terjadi, dan hasil apa yang dihasilkan? (Dharwiyanti dan Wahono, 2003).

**Tabel II.3** Simbol *Sequence Diagram*

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Aktor memainkan bagian dari entitas yang dapat berinteraksi dengan sistem tetapi tidak benar-benar ada di luarnya. Aktor-aktor ini mampu berinteraksi dengan perangkat keras dan perangkat lunak.

2		<i>LifeLine</i>	Eksekusi item dalam seri dilambangkan dengan simbol ini. Dengan kata lain, <i>lifeline</i> adalah komunikasi yang disampaikan dan diterima yang menggambarkan suatu tugas.	
3	<table border="1" data-bbox="459 577 676 651"> <tr> <td>Nama_objek</td> </tr> </table>	Nama_objek	<i>Object</i>	Mengidentifikasi objek yang berkomunikasi dengan pesan.
Nama_objek				
4		<i>Message</i>	Informasi tentang tindakan yang terjadi terkandung dalam spesifikasi komunikasi antar objek.	
5		<i>Message</i>	Informasi tentang tindakan yang terjadi terkandung dalam spesifikasi komunikasi antar objek..	
6	I:keluaran 	Pesan tipe <i>return</i>	Arah panah menunjukkan ke objek yang menerima pengembalian ketika dikatakan bahwa objek yang telah menyelesaikan operasi atau metode memberikan pengembalian ke objek tertentu.	
7	I:Masukan 	Pesan tipe <i>send</i>	Menyatakan bahwa item mengirimkan data, input, atau informasi ke objek lain, arah panah menunjuk ke arah objek yang ditransmisikan	

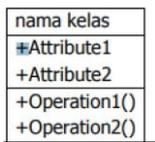
8	I:nama_metode →	Pesan tipe <i>call</i>	Mendeklarasikan objek untuk digunakan sebagai <i>callback</i> untuk tindakan atau metode yang ada baik pada objek tersebut maupun objek lain.
---	--------------------	------------------------	---

Sumber: Rosa A.S dan M. Shalahuddin, 2013

#### 4. Class Diagram

Inti dari desain dan pengembangan berorientasi objek adalah *class*, yang merupakan spesifikasi yang, ketika dipakai, menciptakan objek. *Class* menawarkan layanan untuk mengubah keadaan sistem (metode / fungsi) dan menentukan atribut dan propertinya. *Class diagram* menunjukkan bagaimana kelas, paket, dan objek diatur dan dijelaskan, serta bagaimana mereka berhubungan satu sama lain melalui kurungan, pewarisan, asosiasi, dan interaksi lainnya. (Dharwiyanti dan Wahono, 2003).

**Tabel II.4** Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Class</i>	Pada struktur sistem <i>class</i> , setiap dari <i>class</i> memiliki nama, <i>attribute</i> , dan <i>operation</i> atau <i>method</i> .
2		<i>Nary Association</i>	Upaya untuk mencegah koneksi dengan lebih dari dua item.
3		<i>Association</i>	Hubungan umum antar kelas.
4		<i>Generalization</i>	Generalisasi-spesialisasi (umum-khusus) hubungan antar kelas.
5		<i>Dependency</i>	Hubungan antar kelas yang memperhitungkan saling ketergantungan mereka.

6		<i>Agregasi</i>	Hubungan kelas memiliki makna bagian penuh.
---	---	-----------------	---

Sumber: Rosa A.S dan M. Shalahuddin, 2013

### II.1.6 Android Studio

Pada Google I/O 2013, Google meluncurkan Android Studio, sebuah IDE untuk pengembangan Android. Eclipse IDE, yang didasarkan pada Java IDE IntelliJ IDEA yang populer, adalah tempat Android Studio dikembangkan. (Adibhadiansyah, 2016). IDE resmi untuk mengembangkan aplikasi Android disebut Android Studio (Android Studio, 2016). Pengertian lainnya menurut Bhagat dkk (2018) yaitu lingkungan pengembangan terintegrasi resmi sistem operasi Android, yang disebut Android Studio, dibuat oleh Google dan didasarkan pada program Intelteellij IDEA Jet Brains.

### II.1.7 Website

Situs *web* terdiri dari sejumlah halaman yang ditautkan ke file yang relevan. *Homepage* adalah halaman situs web. Ketika seseorang mengunjungi situs *web*, halaman pertama yang mereka lihat adalah beranda (Turban dalam Jimi, 2020).

Pengguna internet dapat menjelajahi *web* atau melakukan pencarian. Informasi dalam berbagai bentuk atau multimedia dapat disimpan dan disediakan oleh layanan ini. Layanan *web* diinstal pada *web server* sehingga komputer dengan koneksi internet dapat menggunakannya. *Website* atau situs adalah nama untuk tempat di mana data disimpan di *web server* dan dapat dilihat atau diakses oleh pengguna internet (*netter*) di mana-mana (Azhar, 2019).

Sebuah situs terdiri dari satu atau lebih halaman, dengan *homepage* berfungsi sebagai halaman utama. Selain itu, anda dapat "menautkan" seluruh halaman untuk menavigasi dari satu halaman ke halaman lain di situs *web* dan bahkan menautkan ke situs *web global* lainnya. (Azhar, 2019).

### **II.1.8 Bahasa Pemrograman**

Bahasa pemrograman adalah seperangkat instruksi umum untuk mengendalikan komputer. Hal ini juga sering disebut sebagai bahasa komputer atau bahasa pemrograman komputer. Seperangkat aturan sintaksis dan semantik digunakan untuk mendefinisikan program komputer dalam bahasa pemrograman ini. Dengan menggunakan bahasa ini, seorang programmer dapat menentukan data yang diproses komputer, bagaimana data itu disimpan dan dikirimkan, dan tindakan yang dijalankan dalam keadaan tertentu (Saragih, 2016).

Tujuan bahasa pemrograman adalah untuk memberikan instruksi kepada komputer tentang cara memproses data dengan cara yang kita inginkan. Program atau aplikasi adalah apa yang dihasilkan oleh bahasa pemrograman. Contohnya termasuk perangkat lunak yang digunakan kasir di supermarket dan pusat perbelanjaan, penggunaan sinyal lalu lintas di jalan, dll. Menurut Saragih (2016) bahasa pemrograman dibagi menjadi beberapa jenis sesuai dengan tingkat kesulitan dan implementasinya, yaitu:

a. Bahasa Tingkat Tinggi

Karena kemiripannya dengan bahasa manusia, bahasa pemrograman mencapai tingkat ini. seperti PHP, Java, Pascal, Visual Basic, dan Basic.

b. Bahasa Tingkat Menengah

Karena dapat digunakan dalam bahasa tingkat tinggi dan tingkat rendah, itu disebut sebagai tingkat menengah. seperti bahasa pemrograman C.

c. Bahasa Tingkat Rendah

Karena bahasa pemrograman masih sangat berbeda dari bahasa manusia, mereka berada pada level ini. Bahasa Assembly sebagai contoh.

### **II.1.9 PHP**

Menurut Andi (dalam Mubarak, 2019) Bahasa pemrograman yang disebut PHP (*Hypertext Preprocessor*) digunakan sebagai pengolah data di server dan beroperasi di *web server*. *Web server* akan memproses dan menyimpan data pengguna klien dalam *database* di mana ia dapat diambil lagi dan ditampilkan.

Server harus memiliki file yang diunggah agar kode program PHP dapat beroperasi. Tindakan mengunggah data atau file ke *web server* dari komputer klien dikenal sebagai *upload*/unggahan.



**Gambar II.1** Bahasa Pemrograman PHP

(Sumber : jetorbit.com)

Sebuah perangkat lunak yang dapat memproses data dari komputer klien atau dari komputer server sehingga sederhana dan nyaman ditampilkan di *browser* diperlukan untuk mengembangkan situs *web* yang dinamis dan mudah diperbarui setiap saat dari *browser*. PHP adalah program yang dapat digunakan pada server dan sangat dapat diandalkan (Mubarak, 2019).

PHP dapat langsung membuat konten halaman *web* dengan mengintegrasikan dengan dokumen HTML. Alih-alih hanya menjadi kumpulan halaman statis yang jarang diubah, kita dapat menggunakan PHP untuk mengubah situs *web* kita menjadi aplikasi berbasis *web*. (Mubarak, 2019).

PHP pertama kali dimaksudkan untuk bekerja bersama dengan *web server* Apache. Namun, PHP sekarang juga dapat digunakan dengan *web server* seperti Xitami, PWS (*Personal Web Server*), dan IIS (*Internet Information Server*). Dimasukkannya tag menentukan, yang dimulai dengan "<?" atau "<?php" dan diakhiri dengan ">," adalah apa yang membedakan PHP dari bahasa pemrograman lainnya. Dengan demikian kita dapat menempatkan skrip PHP di mana pun dalam file HTML yang kita siapkan. (Mubarak, 2019).

### II.1.10 *Framework* Laravel

*Framework* atau kerangka kerja merupakan suatu struktur konseptual dasar yang dipakai buat memecahkan atau mengatasi suatu kasus yang kompleks. Secara singkat, *framework* merupakan suatu kerangka kerja atau wadah bagi suatu *website* yang akan dibuat. Menggunakan kerangka kerja ini mengurangi waktu yang dibutuhkan untuk membangun situs *web* dan mempermudah untuk melakukan perbaikan (Naista dalam Mediana, 2018).

Sementara Laravel adalah kerangka kerja PHP yang didistribusikan di bawah Lisensi MIT dengan kode sumber yang disediakan oleh Github sebagai referensi, Laravel juga dibundel dengan alat baris perintah yang disebut "Artisan" yang dapat digunakan untuk mengemas bundel dan menginstal bundel dari *command prompt*. Sama seperti kerangka kerja lainnya, Laravel juga dibangun dengan konsep MVC (Model-Controller-View). (Aminudin dalam Yuniyanto, 2021).



**Gambar II.2** *Framework* Laravel

(Sumber : [code.tutsplus.com](http://code.tutsplus.com))

Menurut Aminudin (dalam Yuniyanto, 2021) menyatakan *framework* Laravel memiliki fitur sebagai berikut, antara lain:

a. *Bundles*

*Bundles* adalah karakteristik dari sistem pengemasan modular, dan banyak *bundle* sudah digunakan untuk aplikasi yang ingin dikembangkan.

b. *Eloquent* ORM

Pola "Active Record" adalah implementasi PHP yang canggih, dan *Eloquent* ORM menawarkan cara internal untuk mengelola pembatasan hubungan antara objek database. *Eloquent* dan *Laravel Fluent Query Builder* sangat kompatibel.

c. *Application Logic*

*Application logic* adalah bagian dari aplikasi yang dibuat, baik melalui penggunaan *Controller* atau sebagai komponen deklarasi *Route*. Sintaks definisinya menyerupai kerangka Sinatra.

d. *Reverse Routing*

*Reverse routing* menginterpretasikan hubungan antara *link* dan *route* sehingga ketika *route* berubah, secara otomatis terhubung ke *link* yang berhubungan. Saat membuat tautan menggunakan nama *route* yang ada, Laravel akan secara otomatis menghasilkan URL yang sesuai.

### II.1.11 Flutter

Menurut Tjandra dan Chandra (2020) menyatakan bahwa basis kode tunggal dapat digunakan untuk membangun aplikasi seluler berkinerja tinggi untuk Android dan iOS menggunakan Google Flutter SDK, yang dikembangkan di bawah lisensi *open source*. Tujuannya adalah untuk memberi pengembang alat yang mereka butuhkan untuk menghadirkan aplikasi berkinerja tinggi yang bekerja secara alami di seluruh platform.

Pemahaman tentang Dart diperlukan untuk membangun aplikasi Flutter. Untuk membuat aplikasi lintas platform, Dart adalah bahasa pemrograman khusus. Bahasa pemrograman ini, yang dibuat oleh Google, digunakan dalam pengembangan aplikasi desktop, *server*, *web*, dan seluler. Dart adalah bahasa dengan sintaks C-style yang berorientasi objek dan berbasis kelas. JavaScript atau kode asli keduanya dapat dihasilkan dari Dart. Ini mendukung *generik reified*, tipe inferensi, kelas abstrak, *mixins*, dan antarmuka.



**Gambar II.3** Flutter

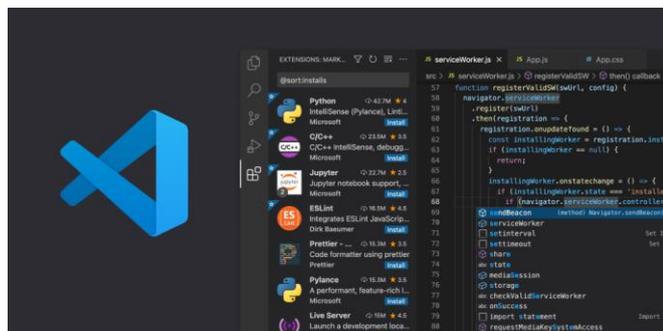
(Sumber : docs.flutter.dev)

Flutter menggunakan konsep *widget* untuk membuat *user interface* penggunanya. Semua *user interface* terdiri dari *widget*. Karena konsep semua *user interface* adalah *widget*, misalnya isi suatu tombol dapat diberikan ke *widget* lain, misalnya dengan memberi gambar, menyetuk dan memberi tombol lagi. Meski hasilnya aneh, Flutter masih bisa menggambar. Menurut Tjandra dan Chandra (2020) ada dua jenis *widget* Flutter:

1. *Stateful widget*, *widget* yang bisa menggambar ulang jika ada perubahan data
2. *Stateless widget*, *widget* yang tidak bisa menggambar ulang jika ada perubahan data

### II.1.12 Visual Studio Code

Program komputer atau perangkat lunak yang memungkinkan pengguna menciptakan, mengubah, atau mengedit file teks (Matondang dkk, 2016). Microsoft membuat editor kode sumber Visual Studio Code untuk Windows, Linux, dan macOS. Dukungan *debugging*, kontrol Git dan GitHub terintegrasi, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, dan pemfaktoran ulang kode sudah disertakan. Tema, pintasan *keyboard*, pengaturan, dan pemasangan ekstensi yang menawarkan fungsionalitas baru semuanya sangat dapat disesuaikan. (Agustini dan Kurniawan, 2020). Kode sumber bahasa pemrograman dapat diedit dan aplikasi komputer dapat ditulis menggunakan Visual Studio Code. Selain itu, dapat digunakan untuk mengembangkan aplikasi tertentu serta halaman web atau template desain web.



**Gambar II.4** Visual Studio Code  
(Sumber : code.visualstudio.com)

### II.1.13 MySQL

Salah satu jenis *database* yang sering digunakan untuk membangun aplikasi *web* dinamis adalah MySQL. Subtipe RDBMS (*Relational Database Management System*) termasuk MySQL. PHP adalah bahasa pemrograman yang didukung oleh MySQL. Mirip dengan PHP, MySQL menawarkan *query* dasar atau bahasa SQL (*Structured Query Language*) dan menggunakan *escape character* yang sama. (Kurniawan dalam Pahlevi dkk., 2018).



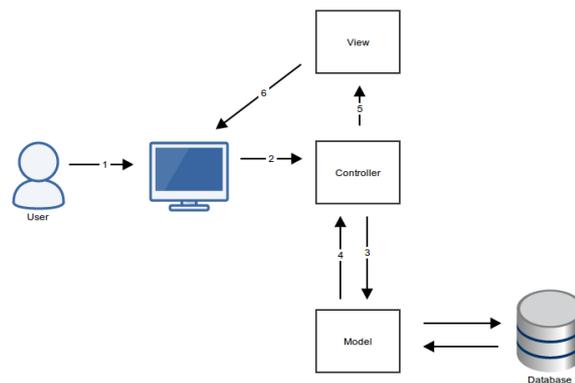
**Gambar II.5** MySQL

(Sumber : id.wikipedia.org)

### II.1.14 MVC (*Model View Controller*)

Pasaribu (2017) menyatakan bahwa pola desain arsitektur yang dikenal sebagai MVC (*Model-View-Controller*) memisahkan aplikasi menjadi tiga komponen logika utama: *model*, *view*, dan *controller*. Bagian-bagian ini semua dibuat untuk menangani aspek-aspek tertentu dari pengembangan aplikasi. Untuk membuat aplikasi besar, terukur, dan dapat diperluas, MVC adalah salah satu kerangka kerja pengembangan *web* standar industri yang paling populer.

Dengan pertumbuhan *framework* PHP seperti Yii, Codeigniter, Laravel, dan lainnya, istilah MVC menjadi lebih terkenal. MVC adalah prinsip yang digunakan oleh hampir semua *framework* PHP ini. *Framework* MVC tidak hanya digunakan dalam PHP, tetapi juga dalam sistem manajemen konten lainnya seperti Joomla, WordPress, Elgg, Prestashop, dan lainnya (Pasaribu, 2017).



**Gambar II.6** Skema MVC

(Sumber : newbienote.com)

Pemrograman dalam gaya MVC membagi logika menjadi tiga kategori: logika *view handling*, logika *controll*, dan logika *model*. MVC berupaya menyederhanakan pemeliharaan untuk pengembangan perangkat lunak yang kompleks.

Adapun komponen MVC antara lain:

1. *Model*

*Query* atau *database* diproses oleh model, yang merupakan bagian dari kode komputer. Dengan demikian, komponen (fungsi) model yang terhubung langsung dengan kemampuan *database* untuk mengedit data, seperti memasukkan data, memperbarui data, menghapus data, dan lain-lain, tetapi yang tidak langsung terkait dengan komponen tampilan.

2. *View*

*View* adalah bagian dari kode perangkat lunak yang mengontrol tampilan situs *web*. Bagian tampilan aplikasi *web* biasanya menggunakan *file template HTML* yang dikontrol oleh *controller*. Meskipun terhubung langsung ke *controller*, wilayah ini tidak memiliki akses langsung ke area *model*. Tampilan dapat disebut sebagai halaman *web* karena mereka dapat menerima dan menampilkan data untuk pengguna.

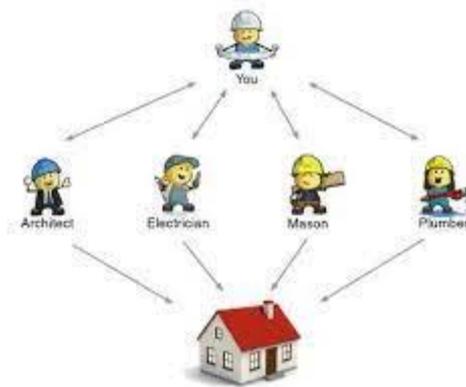
3. *Controller*

Komponen yang menggabungkan model dengan tampilan adalah *controller*. Pemrosesan data dan perintah pengiriman disertakan dalam *controller*.

Tugas *controller* adalah mengambil permintaan dan data pengguna dan memutuskan apa yang harus ditangani aplikasi.

### II.1.15 API (*Application Programming Interface*)

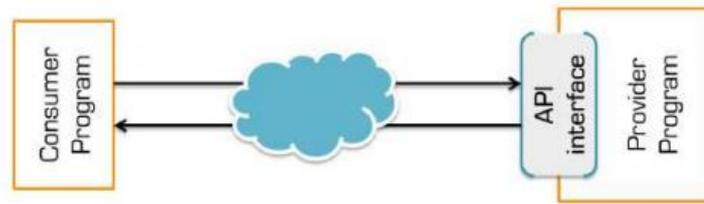
Menurut Reddy (dalam Arianto dkk., 2016) menyatakan bahwa API adalah jenis antarmuka perangkat lunak yang terdiri dari serangkaian instruksi yang disimpan di *library* dan menjelaskan bagaimana satu perangkat lunak dapat berkomunikasi dengan yang lain. Contoh rumah yang akan dibangun dapat digunakan untuk mendemonstrasikan ide ini. Dengan menggunakan kontraktor yang mampu menangani berbagai komponen, pemilik rumah dapat mendelegasikan tugas kepada kontraktor tanpa harus menyadari bagaimana kontraktor menyelesaikan pekerjaan. Dalam contoh ini, rumah mewakili perangkat lunak yang harus dibuat, dan kontraktor mewakili API yang bekerja pada perangkat lunak tertentu tanpa harus terbiasa dengan teknik untuk menyelesaikan tugas.



**Gambar II.7** Analogi API pada Pembangunan Rumah

(Sumber : API Design for C, Reddy, 2011)

*Interface* pada *software* adalah entry point untuk mengakses semua entry points yang terkandung dalam *software*. API menyediakan aturan tentang bagaimana perangkat lunak dapat berinteraksi dengan perangkat lunak lainnya untuk mengakses *resources* melalui *interface* yang tersedia.



**Gambar II.8** Skema Konektivitas API Antar *Software*

(Sumber : What is an API?, 3Scale Networks, 2011)

Spesifikasi struktur data, objek, dan fungsi, serta parameter *input* yang diperlukan untuk mengakses sumber daya dari program, adalah API secara struktural. Antarmuka aplikasi dibentuk oleh semua persyaratan ini, dan API dapat digunakan dengan berbagai bahasa pemrograman atau bahkan hanya dengan menggunakan URL (*Uniform Resource Locators*) yang ditawarkan situs web.

### II.1.16 Pengujian

Pengujian adalah urutan tindakan terorganisir dan metodis untuk memeriksa atau mengevaluasi fakta yang diinginkan (Cholifah dkk., 2018). Adapun pengujian pengujian yang digunakan:

1. *Black Box Testing*, yang menguji perangkat lunak terhadap spesifikasi fungsional daripada desain dan kode program, digunakan untuk memastikan apakah fungsi, input, dan output perangkat lunak sesuai dengan standar yang diperlukan. Dengan pendekatan ini, dimungkinkan untuk menentukan apakah fungsi tersebut masih rentan terhadap data input yang tidak terduga, yang akan membuat data yang disimpan kurang dapat diandalkan. (Cholifah dkk., 2018).
2. *White Box Testing*, disebut juga dengan pengujian struktural. Dimana perangkat lunak yang diuji transparan bagi penguji. Dalam pengujian dirancang dari sudut pandang pengembang dikarenakan struktur internal dikenal dengan menguji segala bagian kode yang mampu untuk diuji dengan tujuan menentukan kesalahan logis dari kode sumber perangkat lunak (Praniffa dkk., 2023).

## II.2 Penelitian Terdahulu

Dalam pembuatan tugas akhir ini, pada dasarnya bukan pertama kali dibahas *e-learning*. Namun, beberapa penelitian sebelumnya yang terkait dengan topik yang disajikan telah dilakukan. Berikut ini adalah *review* dari penelitian-penelitian sebelumnya yang terkait dengan topik penelitian.

**Tabel II.5** *State Of The Art*

No	Nama Peneliti	Judul Penelitian	Tahun Penelitian	Metode Penelitian	Hasil Penelitian
1	Heri Susanto, Tjatarsari Widiartin, dan Firman H.S.P.	Aplikasi Pembelajaran Berbasis Android (E-Learning) Di MA. Daruttaqwa Gresik	2016	Menggunakan metode penelitian model <i>waterfall</i> .	<i>E-Learning</i> MA.Daruttaqwa telah memberikan kemudahan bagi siswa dan guru untuk mendapatkan materi, memberikan materi, menerima tugas, menyerahkan tugas, forum diskusi, dan mendukung proses belajar mengajar secara <i>online</i> . MA.Daruttaqwa <i>E-learning</i> merupakan salah satu alternatif

					perkembangan teknologi yang sebelumnya hanya dapat diakses melalui <i>website Moodle</i> dan kini dapat diakses melalui <i>smartphone Android</i> .
2	Wahyu Eko Susanto dan Yoanna Galuh Ayu Astuti	Perancangan E-Learning Berbasis Web Pada SMP Negeri 3 Patuk Gunungkidul Yogyakarta	2017	Teknik awal pengumpulan data dan pendekatan <i>Waterfall</i> pengembangan perangkat lunak adalah metodologi yang digunakan dalam penelitian ini.	Menurut temuan penelitian, <i>e-learning</i> dapat digunakan sebagai alat untuk meningkatkan proses belajar mengajar antara guru dan siswa. Ini dapat diakses dari lokasi mana saja kapan saja dan dapat meningkatkan tingkat kontak antara guru dan siswa.
3	Arie Krisnoanto	Penerapan Metode	2018	Menggunakan metode UCD	Metodologi <i>User-Centered</i>

		User Centered Design Pada Aplikasi E-Learning Berbasis Android (Studi Kasus: SMAN 3 Sidoarjo)		( <i>User Centered Design</i> )	<i>Design</i> digunakan untuk mengimplementasikan aplikasi pada <i>smartphone</i> Android yang mendukung kegiatan belajar atau <i>e-learning</i> sekolah. Metode ini mengacu pada <i>experience User</i> sehingga dapat membuat aplikasi <i>e-learning</i> yang memiliki nilai kegunaan yang cukup baik untuk digunakan oleh siswa dan guru.
4	Dani Azhar	Aplikasi E-Learning Siswa Berbasis Web Pada SMPN Bernas Kabupaten	2019	Metode <i>waterfall</i> , disebut <i>waterfall</i> karena diagram tahapan prosesnya	Aplikasi <i>e-learning</i> untuk siswa berbasis <i>web</i> telah dibuat di SMPN BERNAS Kab. Pelalawan Riau, data siswa, guru,

		Pelalawan Riau (Studi Kasus: SMPN Bernas Kab. Pelalawan Riau)		mirip dengan air terjun yang bersusun.	mata pelajaran, kelas, tugas, dan sumber daya semuanya termasuk dalam sistem ini. Selain itu, output aplikasi e-learning ini dapat memberikan data nilai berdasarkan pertanyaan yang dijawab dan juga dibangun sesuai dengan rancangan yang ada.
5	Dinda Moulidyna H., Agus Nugroho, dan Abdul Harris	Perancangan E-Learning Pada SMK Veteran Kota Jambi Berbasis Android	2020	Penelitian ini menggunakan Metode <i>Waterfall</i> sebagai metodologinya .	<i>E-learning</i> yang dibangun dapat digunakan untuk membuat soal latihan dan mengunggah materi dan video pembelajaran. <i>Database</i> MySQL dan bahasa pemrograman

					Java digunakan dalam pembuatan <i>e-learning</i> ini.
6	Jimi Asmara	Perancangan Sistem E-Learning Berbasis Web Pada Smp N 2 Busalangga	2020	Metodologi penelitian ini adalah kualitatif. Hasil akhirnya adalah pembuatan konten digital untuk situs <i>web e-learning</i> yang akan dimanfaatkan untuk kegiatan belajar mengajar berbasis IT.	Hasil dari penelitian ini dapat mempermudah kegiatan belajar mengajar guru dan siswa, latihan soal dan mengatasi kendala dalam kegiatan proses belajar mengajar dan implementasi <i>e-learning</i> di SMP Negeri 2 Busalangga. Guru juga dapat masuk ke sistem untuk mengunggah berita, materi pelajaran, artikel dan soal latihan. Sementara itu, siswa dapat melihat artikel

					berita dan pengumuman serta mengunduh materi.
7	Tita Tanjung Sari dan Anang Hadi Cahyono	Pengembangan E-Learning Berbasis Android “Fun Math” Sebagai Alternatif Belajar Matematika di Tengah Pandemi	2020	Penelitian ini merupakan penelitian pengembangan ( <i>research and development</i> ). Tujuan utama pada penelitian ini adalah untuk mengembangkan produk pembelajaran matematika berbasis Android “FunMath”	Dapat dikatakan bahwa “FunMath” valid dan dapat digunakan. “FunMath” juga mendapat tanggapan positif dari siswa yang sudah menggunakan “FunMath” untuk belajar di rumah.
8	Rizky Fauzi Achman, Wina Witanti, Asep Id Hadiana	Pembangunan Perangkat Lunak E-Learning dalam Kegiatan Belajar	2021	Pendekatan Model SDLC, juga dikenal sebagai “waterfall” atau “alur hidup klasik,” kadang-kadang	Dapat berkontribusi dalam proses Kegiatan Belajar Mengajar (KBM) di SMK Angkasa Husein Sastranegara

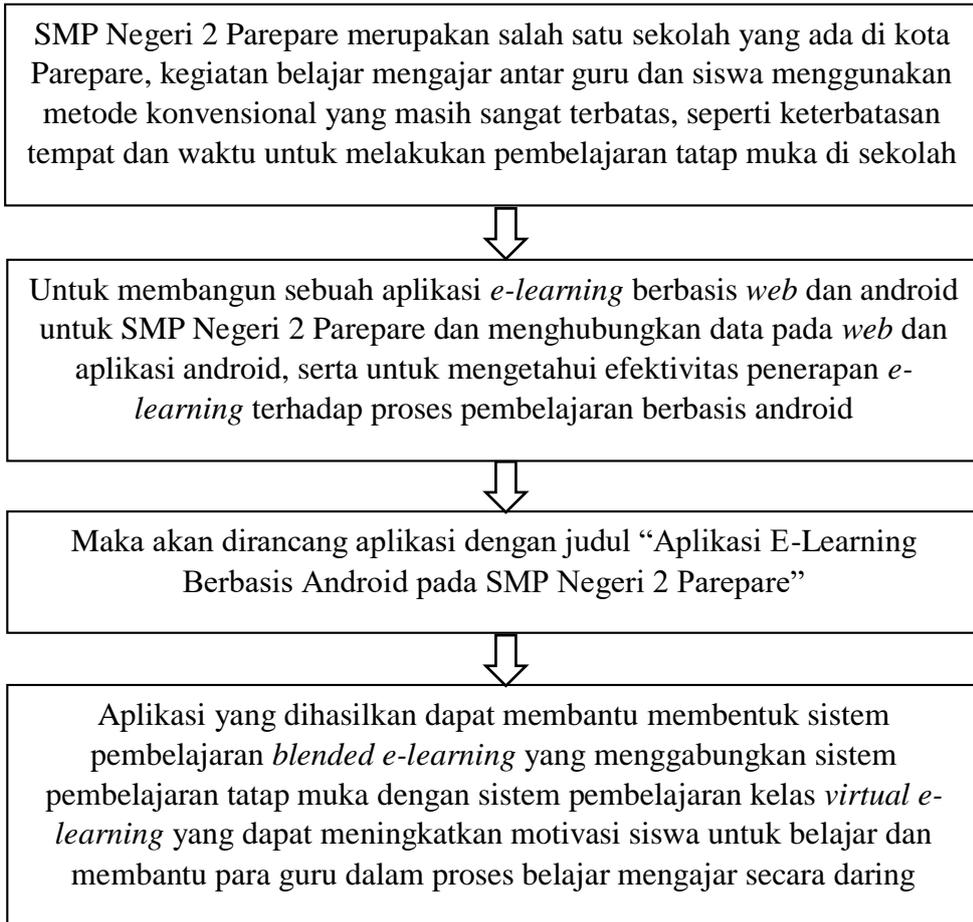
		Mengajar Sekolah Menengah Kejuruan Angkasa Husein		disebut sebagai model sekuensial linier atau "linier berurutan". Pendekatan sekuensial atau sekuensial terhadap siklus hidup perangkat lunak disediakan oleh model air terjun, dimulai dengan analisis persyaratan perangkat lunak, desain, produksi kode program, pengujian, dan pemeliharaan.	Bandung, sehingga lebih cocok untuk KBM.
9	Stiawan Saniago dan Manasse Siahann	Sistem Informasi Akademik Berbasis Web Menggunakan	2021	SDLC ( <i>Software Development Life Cycle</i> ) adalah teknik yang digunakan.	Framework CodeIgniter ini digunakan oleh sistem informasi akademik berbasis web yang dapat

		<p>Framework Codeigniter (Studi Kasus SMA Fatahilih Sidoharjo Jati Agung, Lampung Selatan)</p>		<p>Dimulai dengan langkah-langkah Analisis, Desain, Kode, Pengujian, dan Dukungan, <i>Waterfall</i> menawarkan pendekatan linier untuk siklus hidup perangkat lunak..</p>	<p>menawarkan data akademik berupa data jadwal kelas, nilai, rapor, data guru, dan data siswa. Karena data disimpan dalam database sistem, aman untuk penyimpanan data dan lebih mudah untuk menemukan data. Sistem informasi akademik ini memudahkan untuk membantu siswa dalam mendapatkan informasi akademik dengan cepat dan mudah digunakan, terutama sekolah dalam menyediakan informasi</p>
--	--	--	--	---	--

					akademik dan proses penyimpanan data.
10	Vincentius Abdi Gunawan, Eli Karliani, Triyani, Ahmad S., dan Leonardus S.A.P.	Desain Fitur Aplikasi E-Learning Penunjang Pembelajaran Berbasis Android	2021	Metode yang digunakan menggunakan diagram alir <i>Waterfall</i>	Studi ini menunjukkan hasil dari menciptakan fitur yang menarik bagi siswa dan antarmuka pengguna untuk program pembelajaran. Bagi siswa untuk mendukung media pembelajaran, desain aplikasi pembelajaran sudah menarik.

### II.3 Kerangka Pemikiran

Berikut adalah tahapan-tahapan dalam melakukan penelitian ini yang akan diwujudkan dalam bentuk kerangka pemikiran sebagai berikut:

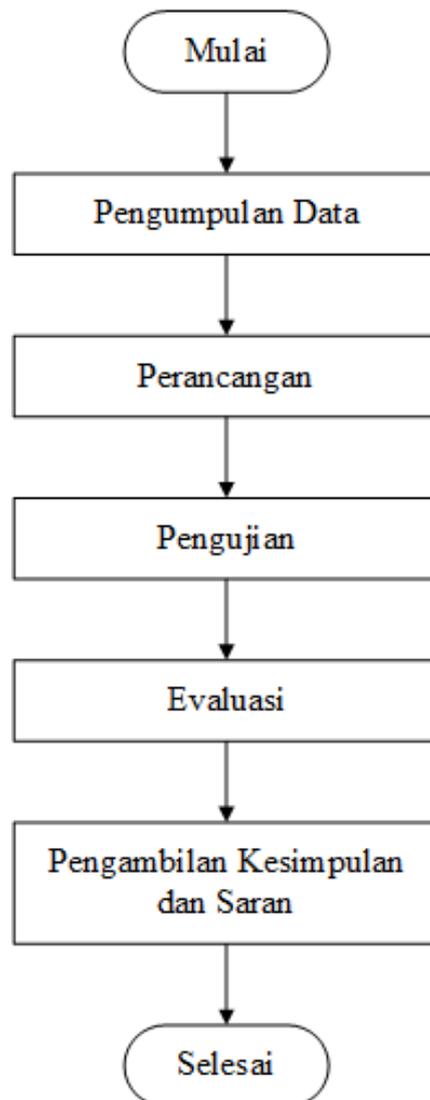


**Gambar II.9** Kerangka Pemikiran

## BAB III METODE PENELITIAN

### III.1 Bagan Alur Penelitian

Beberapa tahap yang akan dilakukan dalam penelitian ini adalah sebagai berikut.



**Gambar III.1 Diagram Alur Penelitian**

Penjelasan diagram alur penelitian pada gambar III.1 adalah sebagai berikut:

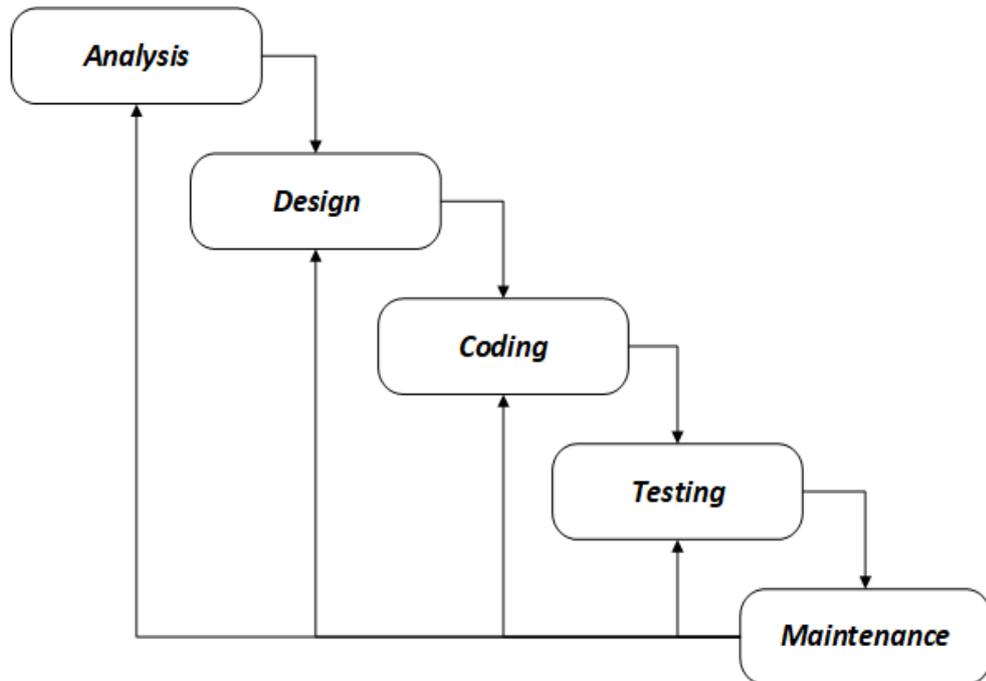
1. Pengumpulan data, dilakukan dengan cara melakukan wawancara terhadap beberapa siswa dan guru, observasi langsung dilapangan, serta

penelusuran jurnal/buku/*ebook*/ dalam rangka mencari informasi yang relevan dengan penelitian.

2. Perancangan, desain aplikasi dilakukan sejalan dengan analisis sistem yang sebelumnya disarankan dan didasarkan pada pemodelan UML (*Unified Modeling Language*), yang meliputi use case diagram, *activity diagram*, *sequence diagram*, dan *class diagram*.
3. Pengujian, Untuk memastikan aplikasi yang dikembangkan dapat berfungsi sesuai rencana, pengujian dilakukan. Semua kelemahan sistem akan ditemukan selama pengujian, bersama dengan apa yang perlu diubah ke depan, memungkinkan hasilnya digunakan sebagai dasar untuk evaluasi, kesimpulan, dan saran.
4. Evaluasi, Pengujian fungsionalitas dan penggunaan sistem disebut evaluasi. Evaluasi dilakukan untuk memverifikasi apakah hasil desain dari prosedur pengujian sistem yang telah dibuat sesuai dengan tuntutan atau tujuan pengguna. Sehingga kita dapat menentukan seberapa baik sistem yang kita rancang akan bekerja dan mungkin menyelesaikan masalah yang melekat.
5. Pengambilan kesimpulan dan saran, dilakukan setelah selesai melakukan seluruh tahapan perancangan aplikasi, pengujian aplikasi dan sudah melakukan evaluasi aplikasi. Selanjutnya penulisan saran untuk memperbaiki kesalahan yang muncul dan memperbaiki penulisan dengan memperhatikan perkembangan aplikasi kedepannya.

### **III.2 Metode Pengembangan Sistem**

Model pengembangan sistem yang digunakan adalah *System Development Life Cycle (SDLC) Waterfall*. *SDLC waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari *analysis*, *design*, *coding*, *testing*, dan *maintenance* (Ade dan Novri, 2019).



**Gambar III.2** Metode *Waterfall*

(Sumber: Jurnal penelitian Achman dkk., 2021)

Adapun penjelasan tahapan-tahapan dari model *waterfall* pada gambar III.2 adalah sebagai berikut:

1. Analisis Kebutuhan Perangkat Lunak (*Analysis*)

Melalui proses observasi dan wawancara, persyaratan perangkat lunak diidentifikasi, memungkinkan pengguna untuk memilih jenis perangkat lunak yang diperlukan. Pada titik ini, spesifikasi kebutuhan perangkat lunak harus didokumentasikan.

2. Desain (*Design*)

Struktur data, arsitektur perangkat lunak, desain antarmuka, dan konvensi pengkodean adalah bagian dari multi-proses desain perangkat lunak, yang berfokus pada pembuatan aplikasi perangkat lunak. Agar dapat diimplementasikan dalam program nanti, tahap ini mengubah persyaratan perangkat lunak dari tahap analisis persyaratan menjadi representasi desain.

3. Pembuatan Kode Program (*Coding*)

Desain harus diubah menjadi program komputer. Program komputer yang dibuat sesuai dengan desain yang dibuat dalam tahap desain adalah hasil dari langkah ini. Kode program ini dikembangkan menggunakan bahasa pemrograman PHP (Laravel) dan Flutter untuk Android.

#### 4. Pengujian (*Testing*)

Pengujian memastikan bahwa semua komponen perangkat lunak telah diuji dari sudut pandang logis dan fungsional. Hal ini dilakukan untuk mengurangi kesalahan dan menjamin bahwa hasilnya adalah apa yang diinginkan. Metode pengujian *Black Box Testing*, *White Box Testing*, dan pengujian efektivitas perangkat lunak digunakan untuk menguji sistem ini.

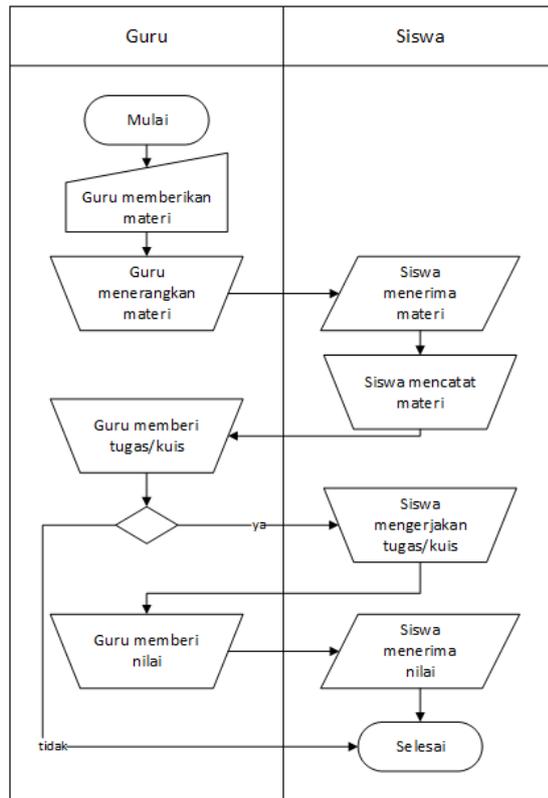
#### 5. Pemeliharaan (*Maintenance*)

Pengembangan sistem dilakukan pada fase ini, termasuk penambahan fitur dan fungsi baru. Namun, fase pemeliharaan tidak termasuk dalam topik penyelidikan penelitian ini.

### **III.3 Rancangan Penelitian**

#### **III.2.1 Sistem yang Berjalan**

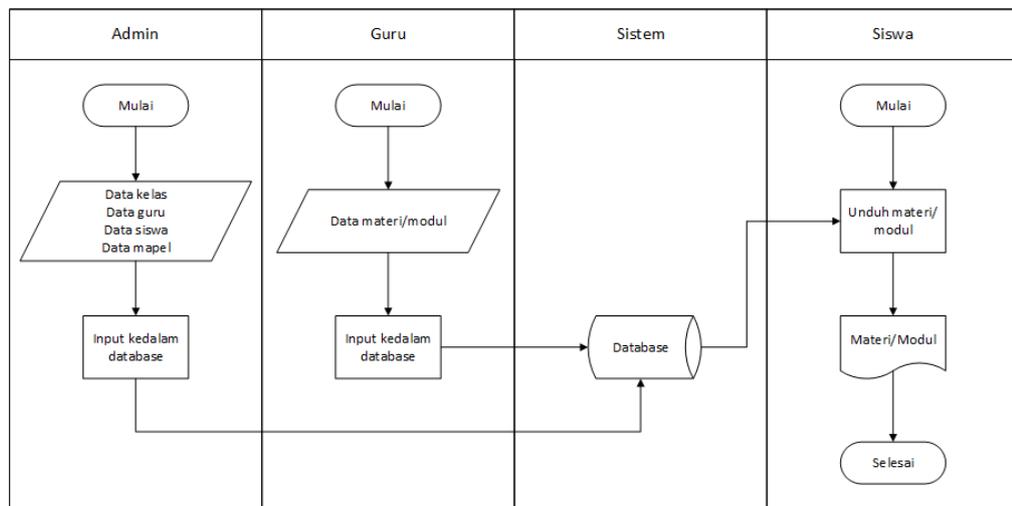
Sistem pembelajaran yang sedang berjalan di SMP Negeri 2 Parepare masih secara konvensional. Pembelajaran konvensional yang dimaksud disini adalah kegiatan ini dilakukan secara tatap muka yang terjadi didalam kelas dan wilayah sekolah pada jam pelajaran yang telah ditetapkan, dimana guru menerangkan materi melalui buku, modul, atau presentasi dengan lcd; siswa menyimak serta mencatat. Guru memberikan tugas kepada siswa dan siswa mengerjakannya; dan diperiksa serta diberi nilai saat itu juga. Pemberian kuis melalui *google form*, dan pemberian ujian menggunakan aplikasi Exambro.



**Gambar III.3** Sistem yang Berjalan

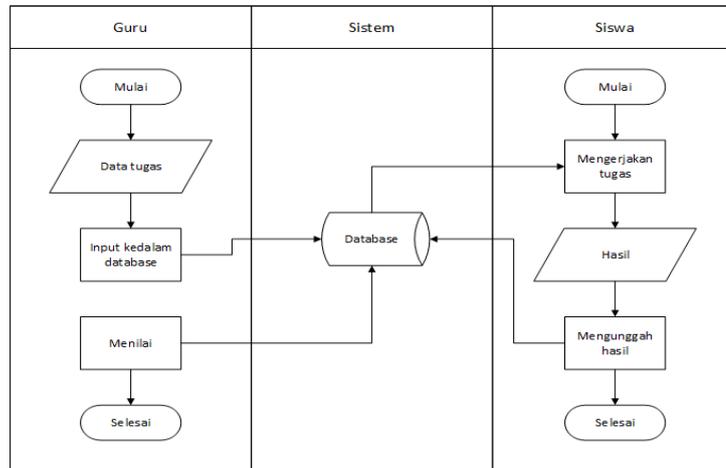
### III.2.2 Sistem yang Diusulkan

Dari analisis sistem yang berjalan, berikut merupakan sistem pembelajaran, pemberian tugas, pemberian kuis, dan pemberian ujian yang diusulkan.



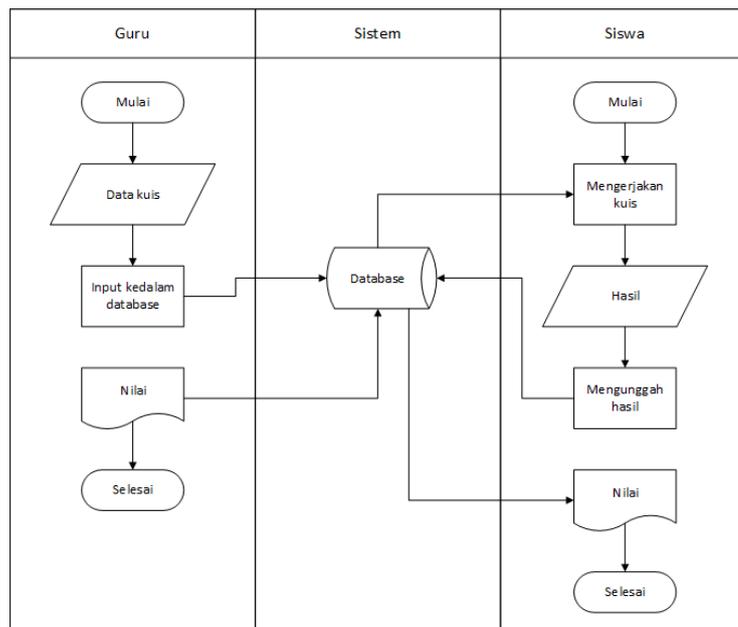
**Gambar III.4** Sistem Pembelajaran yang Diusulkan

Pada gambar III.4, proses pembelajaran yang diusulkan dimana admin dimulai dengan penginputan data guru, data kelas, data siswa, dan data mapel ke dalam sistem. Lalu guru menginput data materi/modul ke dalam sistem yang nantinya siswa dapat mengunduh materi/modul tersebut.



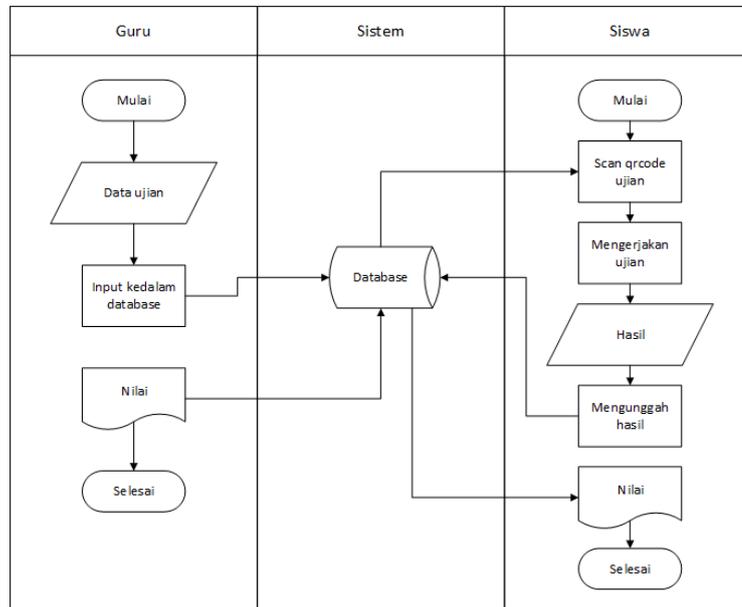
**Gambar III.5** Sistem Pemberian-Pengerjaan Tugas yang Diusulkan

Pada gambar III.5, proses pemberian tugas dapat dilakukan oleh guru yang awalnya dengan menginput tugas ke dalam sistem, dimana nantinya siswa dapat mengerjakan tugas tersebut dan mengunggah hasil dari tugas tersebut ke dalam sistem, dilanjutkan dengan proses penilaian dari guru.



**Gambar III.6** Sistem Pemberian-Pengerjaan Kuis yang Diusulkan

Pada gambar III.6, proses pemberian kuis dapat dilakukan oleh guru yang awalnya dengan *input* data kuis ke dalam sistem, dimana nantinya siswa dapat mengerjakan kuis tersebut dan mengunggah hasil dari kuis tersebut ke dalam sistem, dalam hal ini sistem melakukan penilaian yang berdasarkan ditetapkan guru dalam sistem, dan hasil dari penilaian tersebut akan tampil di guru dan juga siswa.



**Gambar III.7** Sistem Pelaksanaan Ujian yang Diusulkan

Pada gambar III.7, proses pelaksanaan ujian dimulai oleh guru yang awalnya dengan *input* data ujian ke dalam sistem, dimana nantinya siswa dapat memindai *qr code* ujian yang nantinya sebagai absensi ujian dan sebagai fitur untuk memulai ujian, setelah itu siswa dapat mengerjakan ujian dan mengunggah hasil dari ujian tersebut ke dalam sistem, dalam hal ini sistem melakukan penilaian yang berdasarkan ditetapkan guru dalam sistem, dan hasil dari penilaian tersebut akan tampil di guru dan juga siswa.

### III.4 Waktu dan Lokasi Penelitian

Penelitian ini dilakukan di SMP Negeri 2 Parepare. Penelitian ini dilaksanakan pada bulan November 2022–Februari 2023.

### III.5 Alat dan Bahan

Berikut adalah alat dan bahan yang digunakan:

#### 1. Perangkat Keras (*Hardware*)

**Tabel III.1** Perangkat Keras

<b>Nama Komponen</b>	<b>Unit</b>	<b>Spesifikasi</b>
Laptop Asus A442UF	1	Processor Intel Core i5-8250U Installed RAM 12,0 GB Windows 10 Home 64-bit HDD 1 TB
<i>Smartphone</i> Android (Realme 6)	1	Chipset MediaTek Helio G90T RAM 8GB ROM 128GB

#### 2. Perangkat Lunak (*Software*)

Adapun perangkat lunak yang digunakan sebagai berikut:

- a. Visual Studio Code
- b. Bahasa Pemrograman PHP (*Framework* Laravel)
- c. Bahasa Pemrograman Dart (*Framework* Flutter)
- d. Xampp
- e. Android Studio

### III.6 Metode Pengumpulan Data

Untuk melakukan penelitian, pengumpulan data sangat penting. karena dimaksudkan untuk mengumpulkan data yang dapat diandalkan untuk digunakan dalam proses penelitian. Berikut adalah beberapa cara untuk mengumpulkan data:

#### 1. Wawancara

Wawancara adalah jenis pengumpulan data di mana peneliti menanyai informan tentang segala sesuatu dalam upaya untuk mendapatkan informasi yang mereka butuhkan. Teknik wawancara ini

dilakukan buat melengkapi data yang berhubungan pada penelitian. Wawancara dilakukan dengan beberapa guru dan siswa.

2. Observasi

Melakukan observasi dengan cara pengamatan secara langsung di lokasi. Peneliti melakukan pengamatan dengan mengamati dan mendokumentasikan proses belajar mengajar di kelas. Pengamatan ini dimaksudkan untuk berfungsi sebagai panduan untuk pengembangan aplikasi.

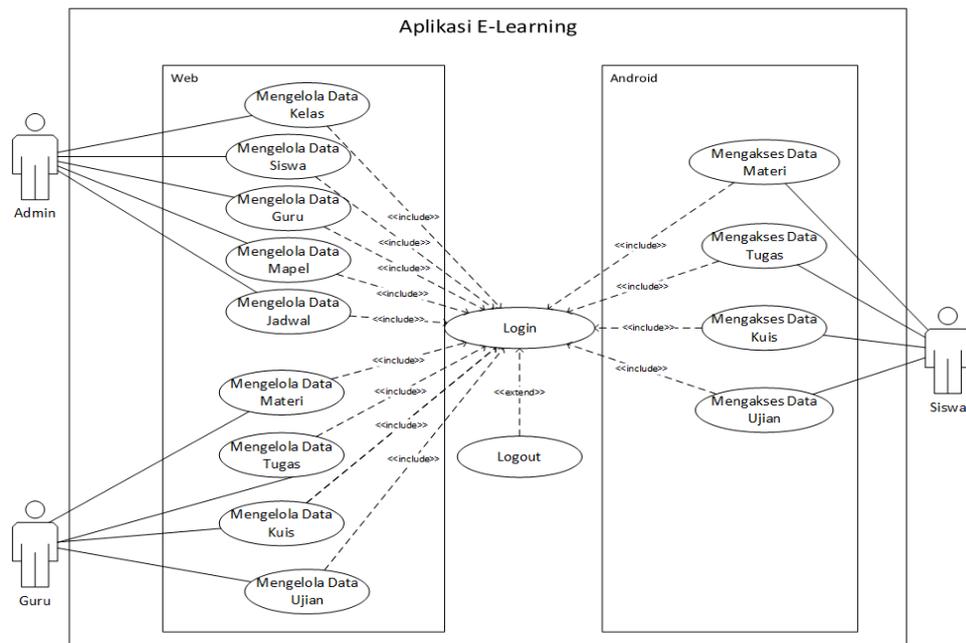
3. Studi Pustaka

Pengumpulan data dilakukan oleh peneliti dengan mencari buku, jurnal, *website*, dan literatur dengan topik pembahasan yang relevan pada penelitian sebelumnya.

**III.7 Analisis Sistem**

Analisis sistem dibuat menggunakan diagram UML (*Unified Modelling Language*)

**III.6.1 Use Case Diagram**

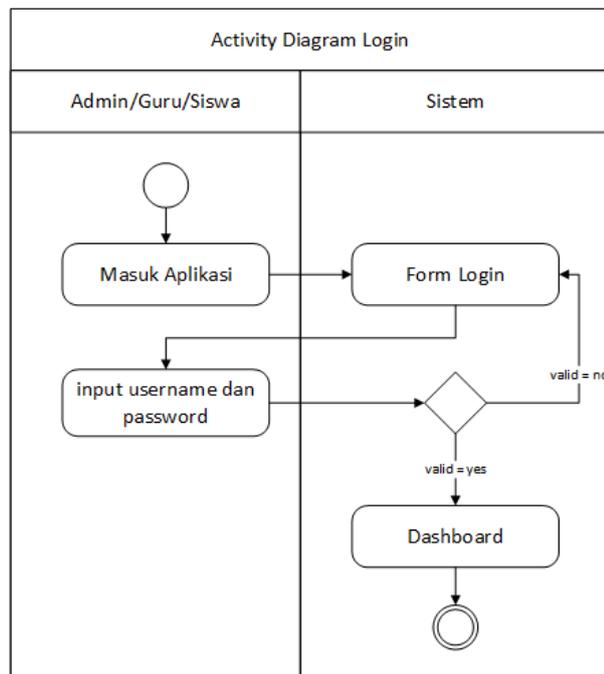


**Gambar III.8 Use Case Diagram**

*Use case diagram* sistem e-learning dapat dilihat pada gambar III.8 menunjukkan fungsi-fungsi yang harus ada dalam sistem. Dalam diagram tersebut terdapat tiga aktor yaitu admin, guru, dan siswa. Pada diagram *use case* untuk aktor admin berhubungan dengan lima *use case* yaitu mengelola data kelas, mengelola data siswa, mengelola data guru, mengelola data mapel, dan mengelola data jadwal. Untuk aktor guru berhubungan dengan empat *use case* yaitu mengelola data materi, mengelola data tugas, mengelola data kuis, dan mengelola data ujian. Untuk aktor siswa berhubungan dengan empat *use case* yaitu mengakses data materi, mengakses data tugas, mengakses data kuis, dan mengakses data ujian. Untuk mengakses semua *use case*, tiga aktor tersebut harus melakukan *login* terlebih dahulu.

### III.6.2 Activity Diagram

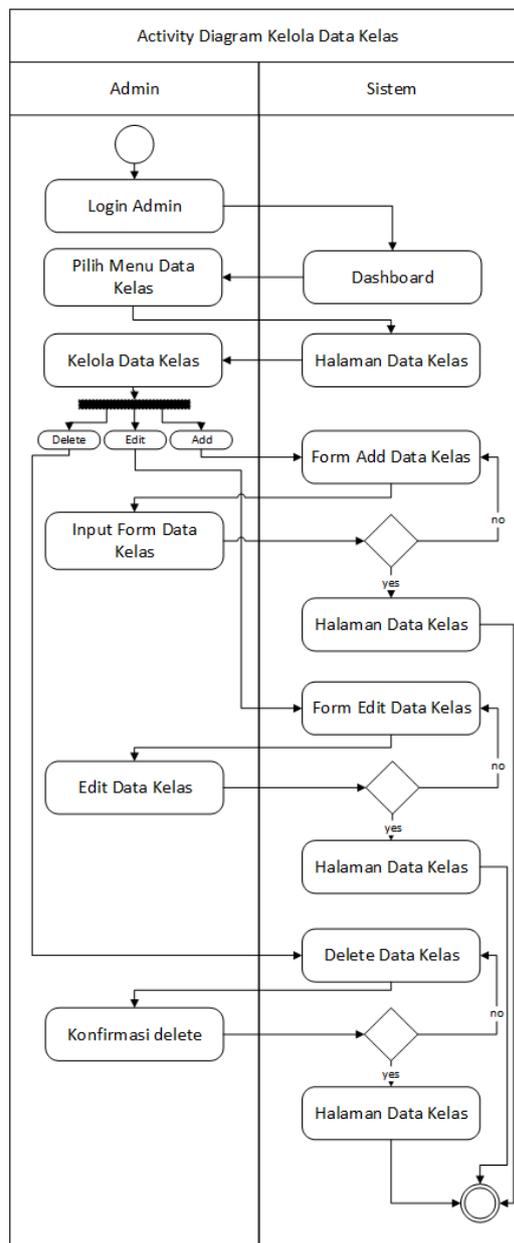
#### 1. Activity Diagram Login Admin, Guru, dan Siswa



**Gambar III.9** Activity Diagram Login

Pada gambar III.9, *activity diagram login* guru, admin, maupun siswa yang diharuskan *login* sebelum mengakses menu-menu yang ada di aplikasi.

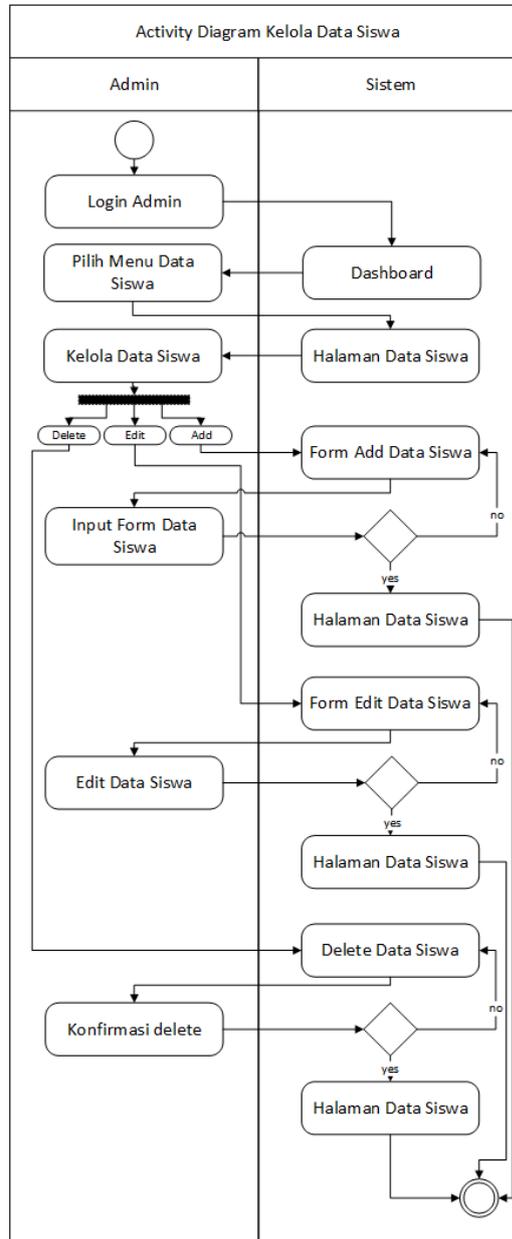
## 2. Activity Diagram Kelola Data Kelas oleh Admin



**Gambar III.10** Activity Diagram Kelola Data Kelas

Pada gambar III.10, *activity diagram* kelola data kelas oleh admin. Dimana admin dapat mengelola data kelas seperti menambah data kelas, mengedit data kelas, serta menghapus data kelas.

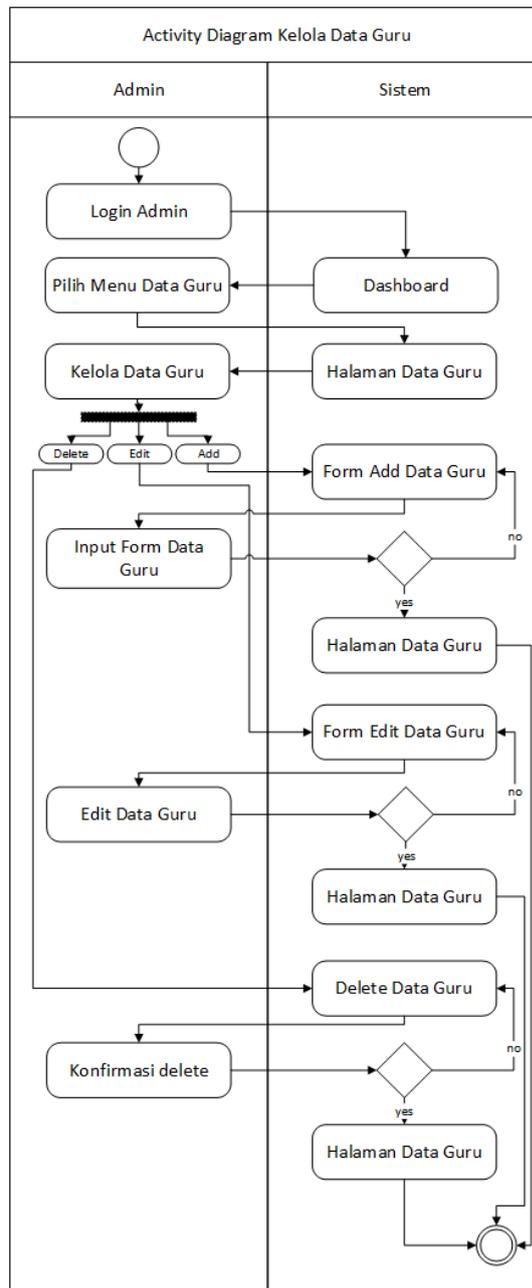
### 3. Activity Diagram Kelola Data Siswa oleh Admin



**Gambar III.11** Activity Diagram Kelola Data Siswa

Pada gambar III.11, *activity diagram* kelola data siswa oleh admin. Dimana admin dapat mengelola data siswa seperti menambah data siswa, mengedit data siswa, serta menghapus data siswa.

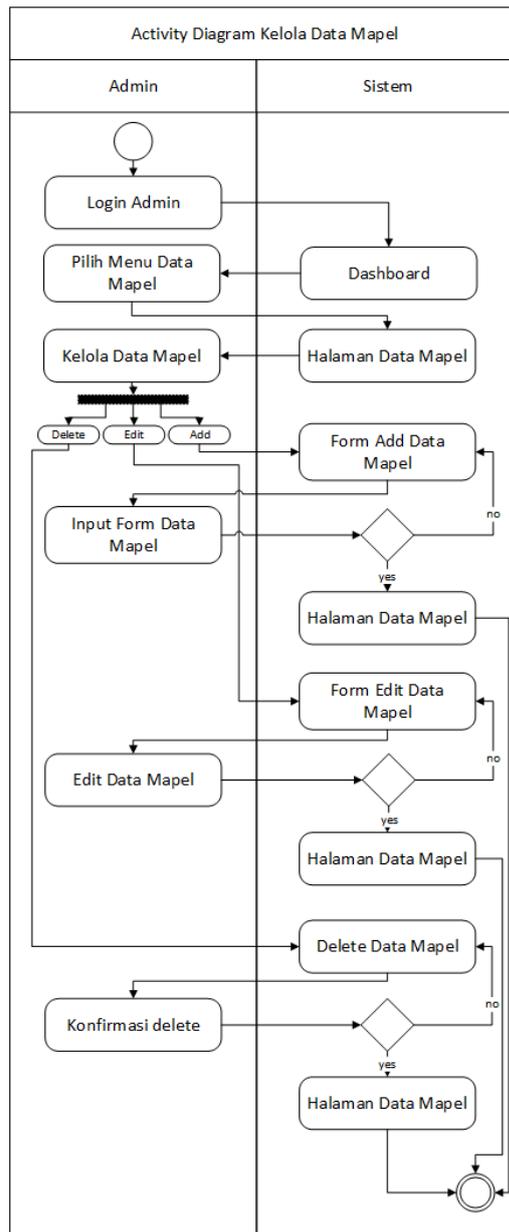
#### 4. Activity Diagram Kelola Data Guru oleh Admin



**Gambar III.12** Activity Diagram Kelola Data Guru

Pada gambar III.12, *activity diagram* kelola data guru oleh admin. Dimana admin dapat mengelola data guru seperti menambah data guru, mengedit data guru, serta menghapus data guru.

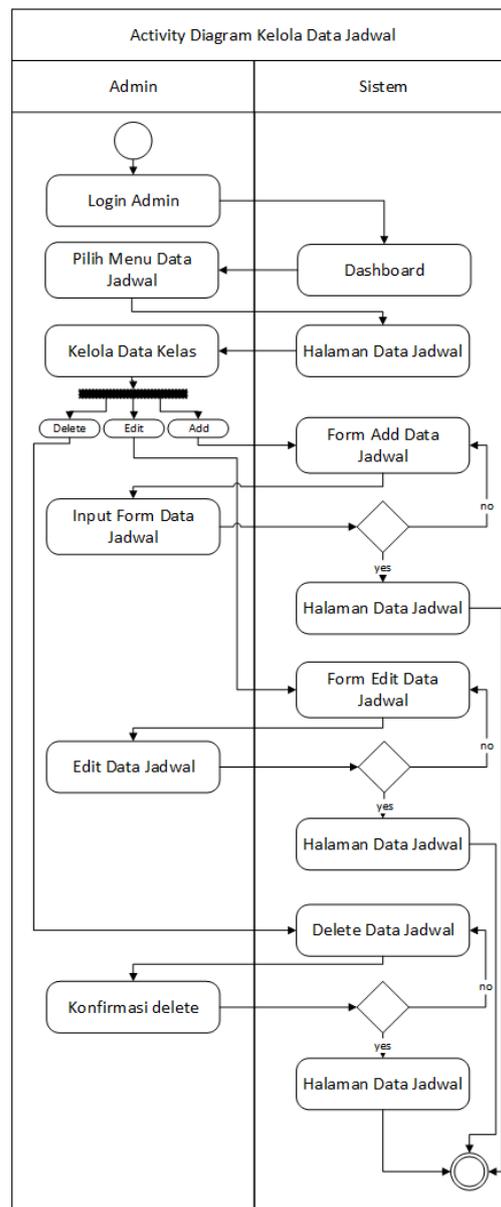
### 5. Activity Diagram Kelola Data Mapel oleh Admin



**Gambar III.13** Activity Diagram Kelola Data Mapel

Pada gambar III.13, *activity diagram* kelola data mapel oleh admin. Dimana admin dapat mengelola data mapel seperti menambah data mapel, mengedit data mapel, serta menghapus data mapel.

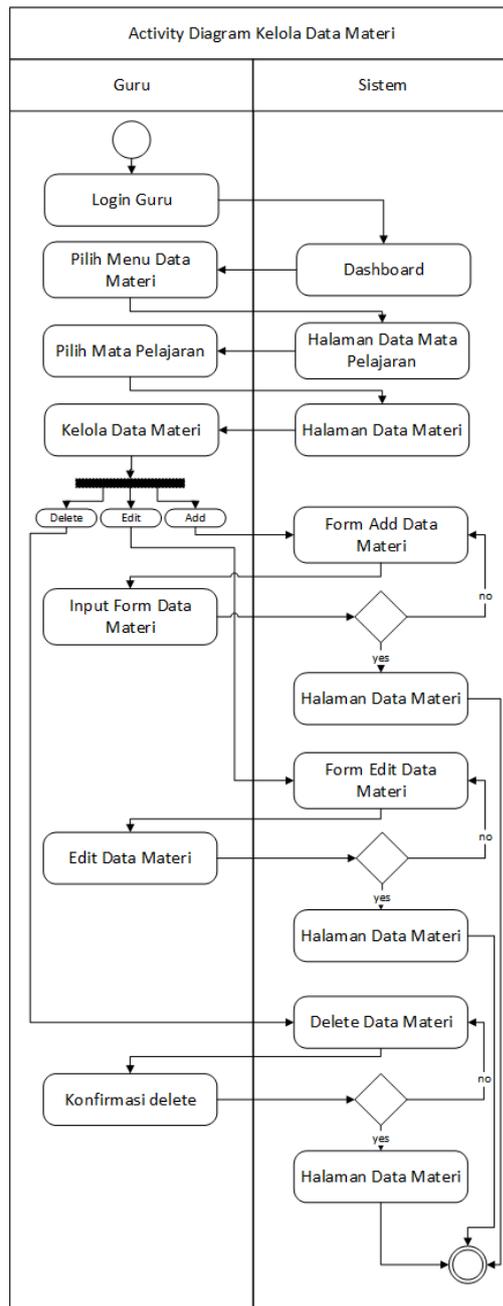
## 6. Activity Diagram Kelola Data Jadwal oleh Admin



**Gambar III.14** Activity Diagram Kelola Data Jadwal

Pada gambar III.14, *activity diagram* kelola data jadwal oleh admin. Dimana admin dapat mengelola data jadwal seperti menambah data jadwal, mengedit data jadwal, serta menghapus data jadwal.

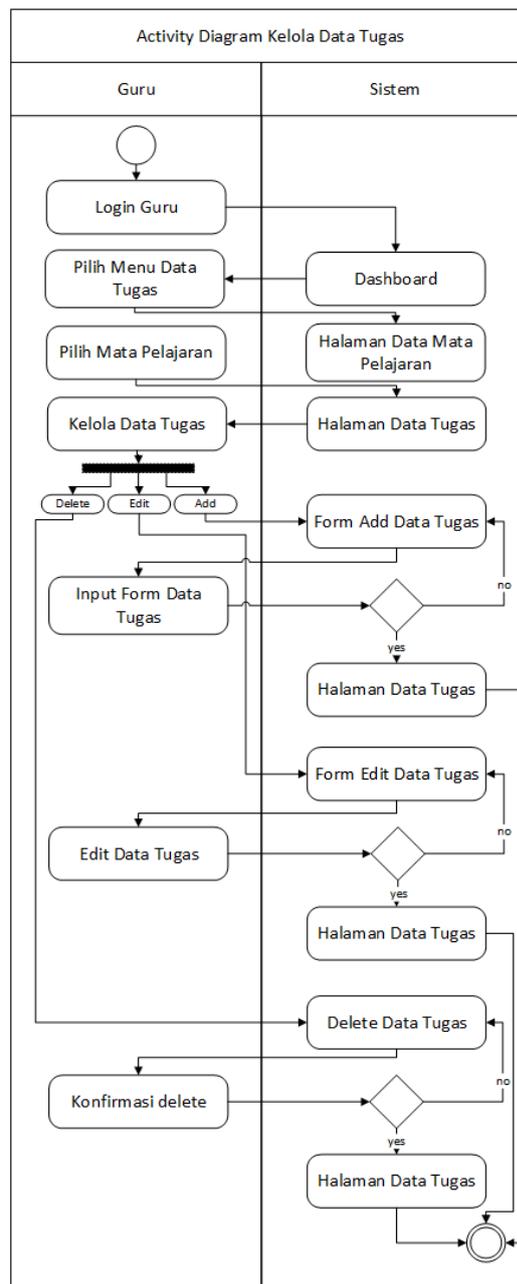
7. Activity Diagram Kelola Data Materi oleh Guru



**Gambar III.15** Activity Diagram Kelola Data Materi

Pada gambar III.15, *activity diagram* kelola data materi oleh guru. Dimana guru dapat mengelola data materi seperti menambah data materi, mengedit data materi, serta menghapus data materi.

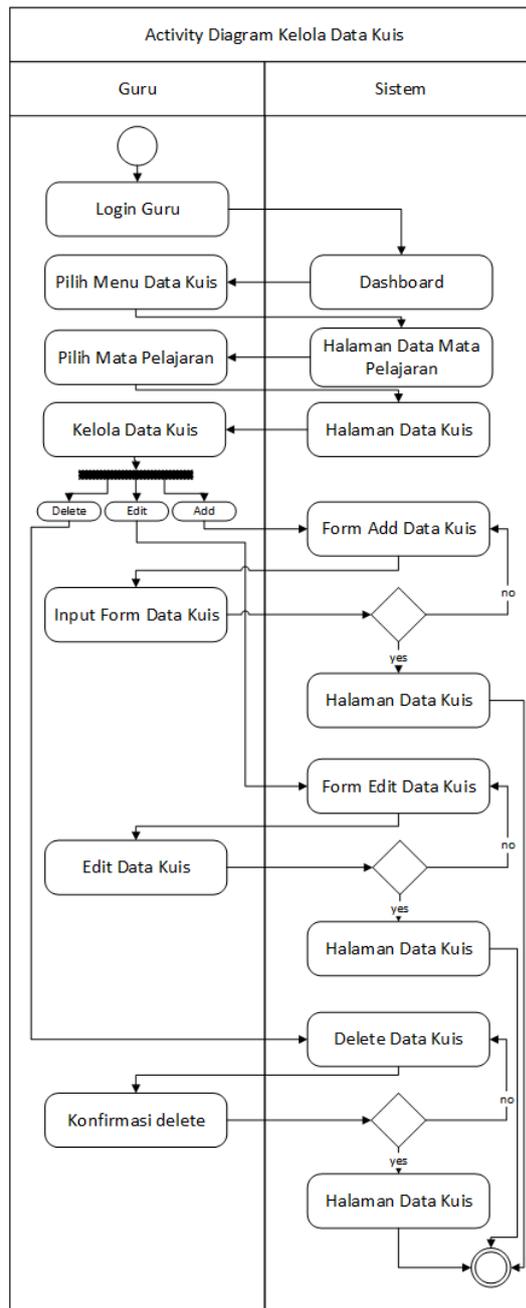
## 8. Activity Diagram Kelola Data Tugas oleh Guru



**Gambar III.16** Activity Diagram Kelola Data Tugas

Pada gambar III.16, *activity diagram* kelola data tugas oleh guru. Dimana guru dapat mengelola data tugas seperti menambah data tugas, mengedit data tugas, serta menghapus data tugas.

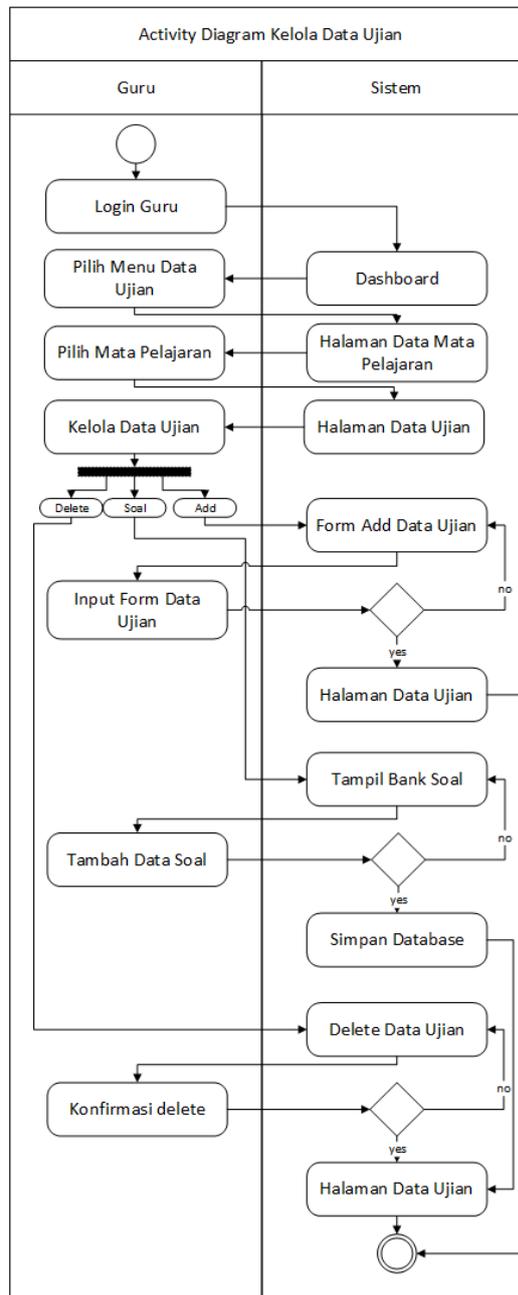
### 9. Activity Diagram Kelola Data Kuis oleh Guru



**Gambar III.17** Activity Diagram Kelola Data Kuis

Pada gambar III.17, *activity diagram* kelola data kuis oleh guru. Dimana guru dapat mengelola data kuis seperti menambah data kuis, mengedit data kuis, serta menghapus data kuis.

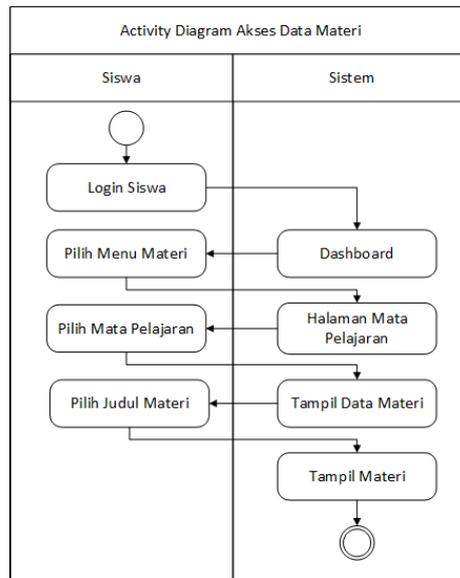
### 10. Activity Diagram Kelola Data Ujian oleh Guru



**Gambar III.18** Activity Diagram Kelola Data Ujian

Pada gambar III.18, *activity diagram* kelola data ujian oleh guru. Dimana guru dapat mengelola data ujian seperti menambah data ujian, mengelola bank soal, serta menghapus data ujian.

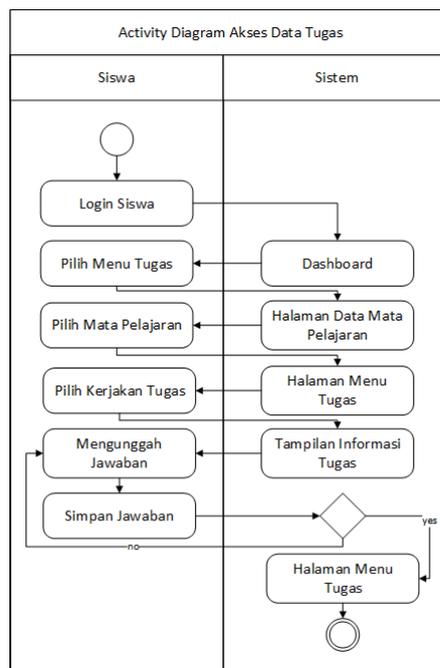
### 11. Activity Diagram Akses Data Materi oleh Siswa



**Gambar III.19** Activity Diagram Akses Data Materi

Pada gambar III.19, *activity diagram* akses data materi oleh siswa. Dimana siswa dapat menampilkan dan melihat materi.

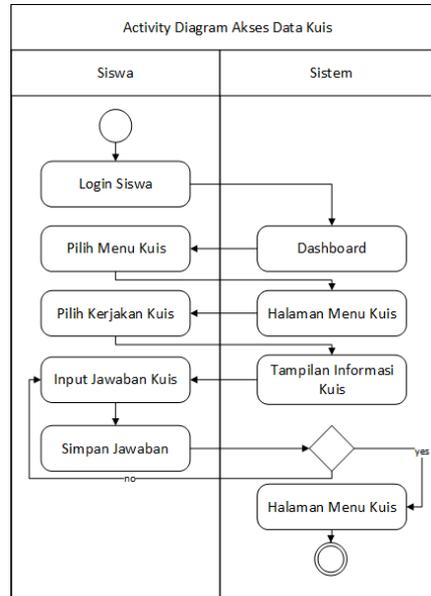
### 12. Activity Diagram Akses Data Tugas oleh Siswa



**Gambar III.20** Activity Diagram Akses Data Tugas

Pada gambar III.20, *activity diagram* akses data tugas oleh siswa. Dimana siswa dapat mengerjakan tugas dengan cara mengunggah jawaban.

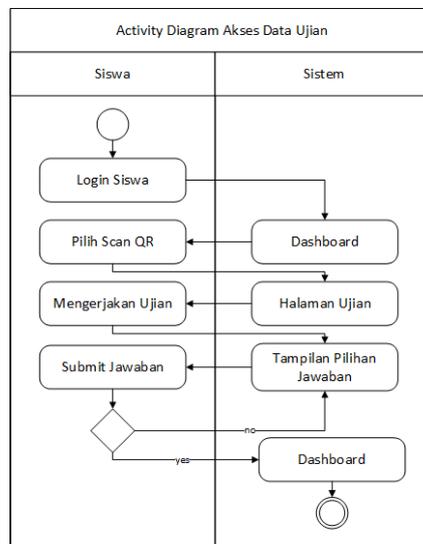
### 13. Activity Diagram Akses Data Kuis oleh Siswa



**Gambar III.21** Activity Diagram Akses Data Kuis

Pada gambar III.21, *activity diagram* akses data kuis oleh siswa. Dimana siswa dapat mengerjakan kuis dengan cara mengisi jawaban kuis.

### 14. Activity Diagram Akses Data Ujian oleh Siswa

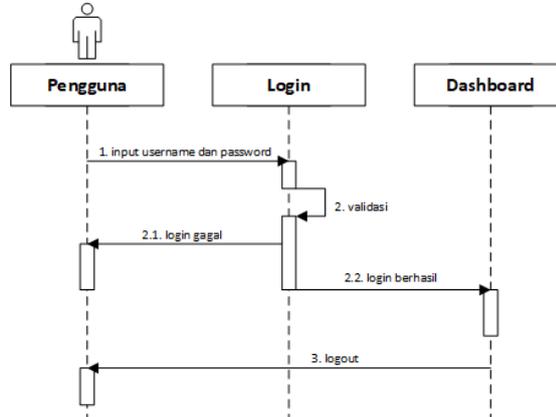


**Gambar III.22** Activity Diagram Akses Data Ujian

Pada gambar III.22, *activity diagram* akses data ujian oleh siswa. Dimana siswa dapat mengerjakan ujian dengan cara memilih jawaban ujian.

### III.6.3 Sequence Diagram

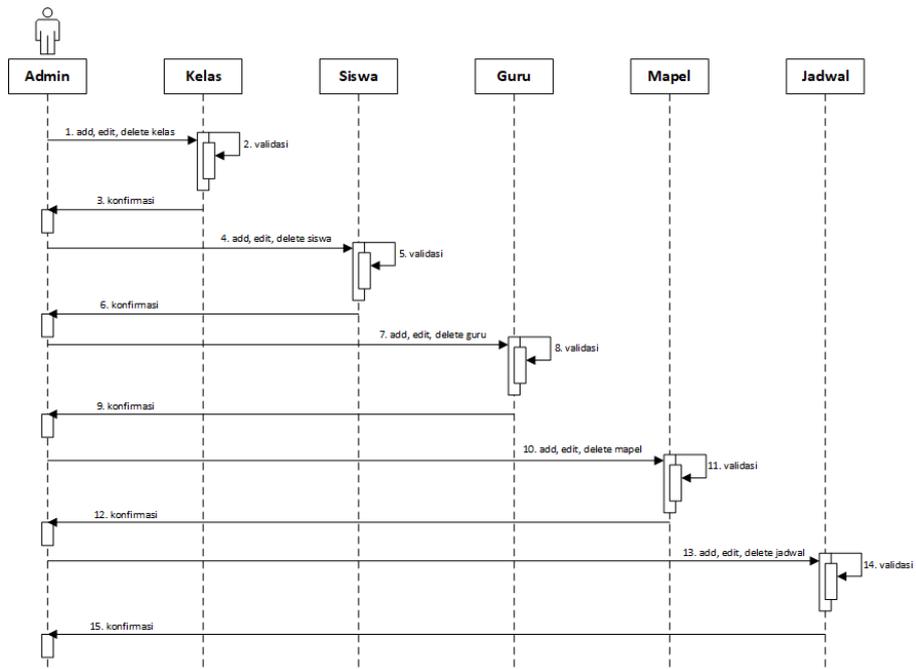
#### 1. Sequence Diagram Login Admin, Guru, dan Siswa



**Gambar III.23** Sequence Diagram Login

Sequence diagram login dimana admin, guru, dan siswa melakukan login sebelum mengakses dashboard.

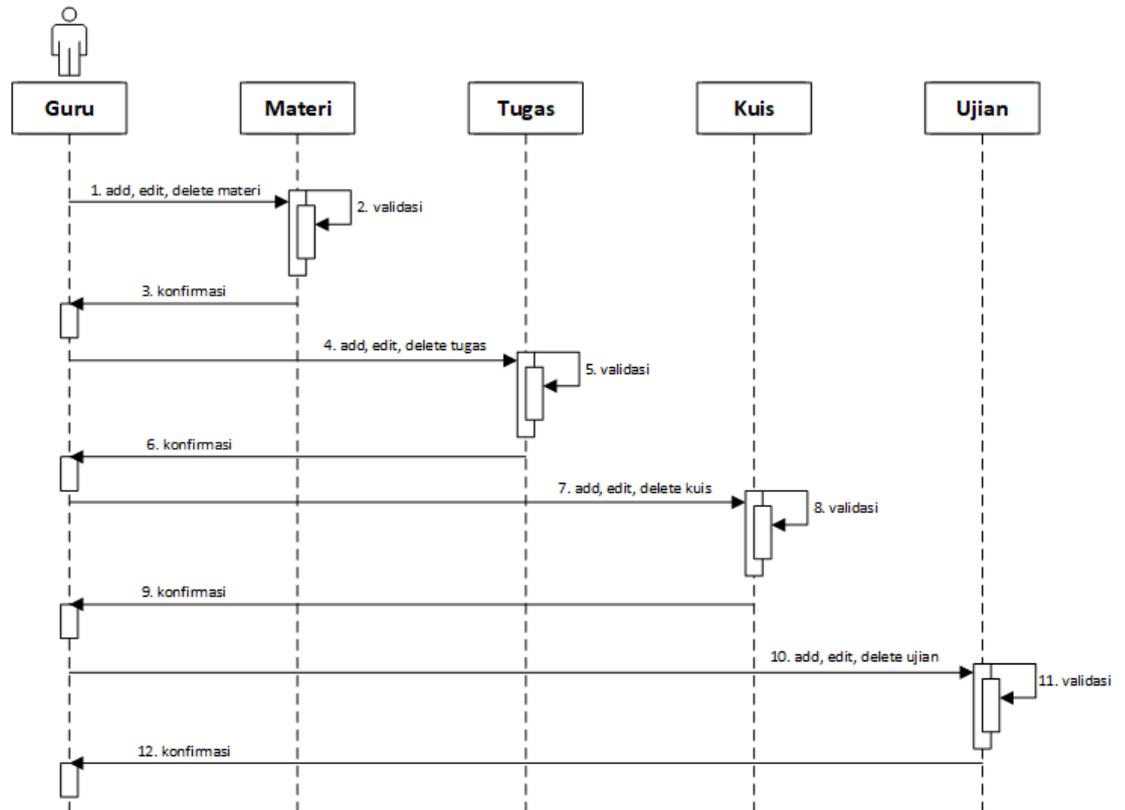
#### 2. Sequence Diagram Kelola E-Learning Admin



**Gambar III.24** Sequence Diagram Kelola E-Learning Admin

*Sequence diagram* kelola menu *e-learning* dimana admin dapat melakukan pengelolaan pada menu data kelas, data siswa, data guru, data mapel, dan data jadwal.

### 3. *Sequence Diagram* Kelola E-Learning Guru

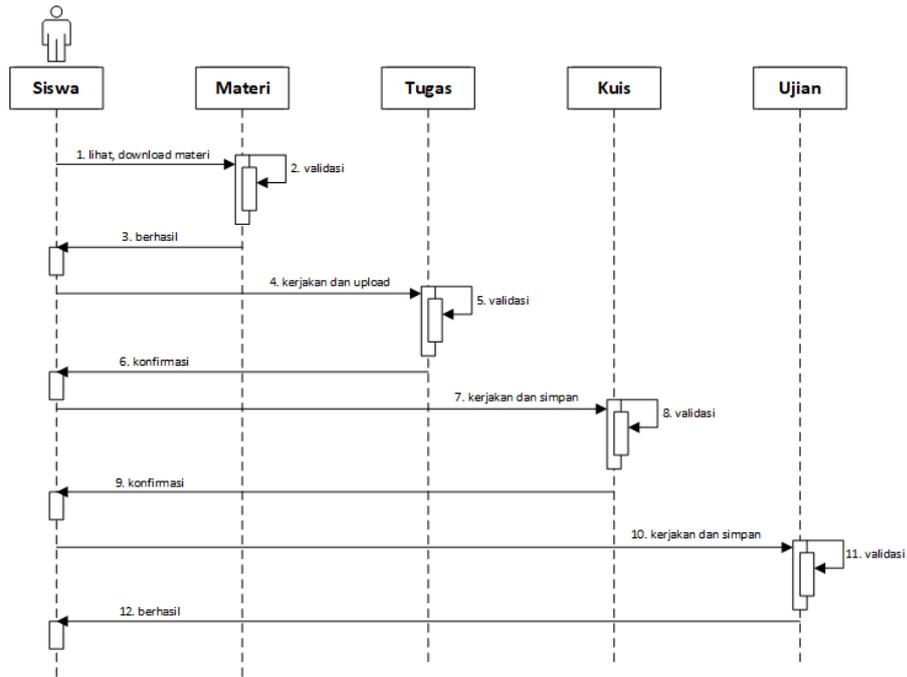


**Gambar III.25** *Sequence Diagram* Kelola E-Learning Guru

*Sequence diagram* kelola menu *e-learning* dimana guru dapat melakukan pengelolaan pada menu data materi, data tugas, data kuis, data mapel, dan data ujian.

### 4. *Sequence Diagram* Akses E-Learning Siswa

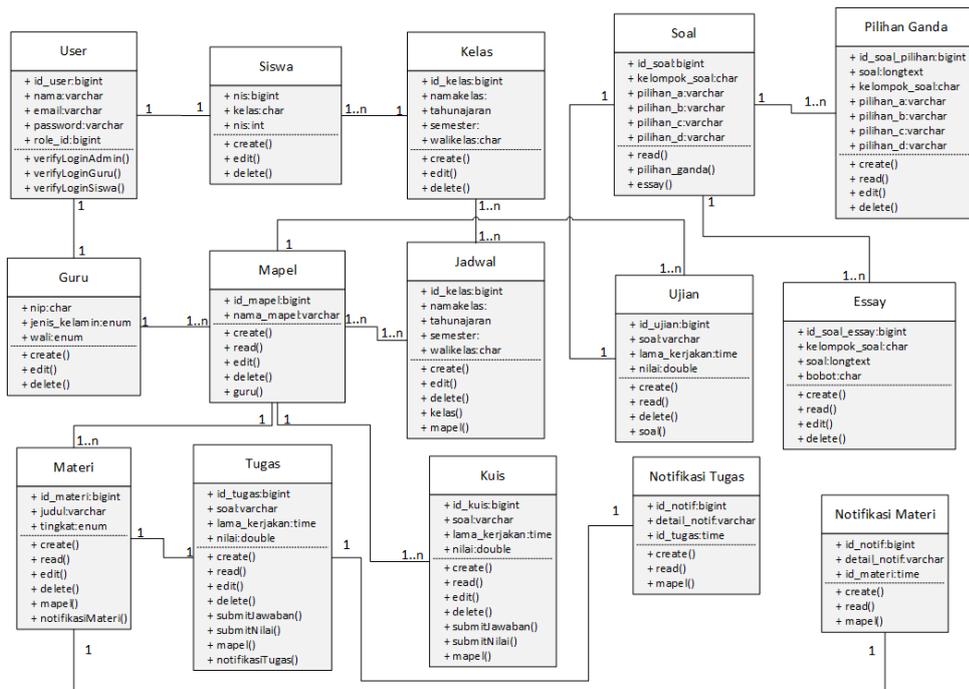
*Sequence diagram* akses *e-learning* dimana siswa dapat mengakses menu-menu yang ada di aplikasi *e-learning* seperti menu materi, tugas, kuis, dan ujian.



Gambar III.26 Sequence Diagram Akses E-Learning Siswa

### III.6.4 Class Diagram

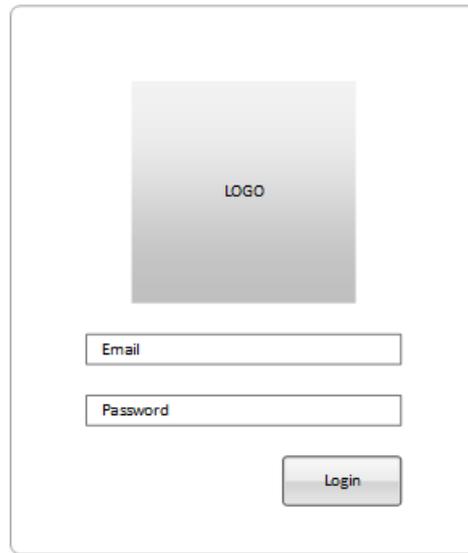
Berikut merupakan *class diagram* yang menjelaskan secara singkat hubungan relasi antar satu *class* dengan *class* lainnya.



Gambar III.27 Class Diagram E-Learning

### III.6.5 User Interface

#### 1. Tampilan *Login Admin dan Guru (Web)*



The image shows a web login interface. At the top center is a square placeholder labeled "LOGO". Below the logo are two input fields: the first is labeled "Email" and the second is labeled "Password". At the bottom right of the form is a button labeled "Login".

**Gambar III.28** *Interface Login Admin dan Guru*

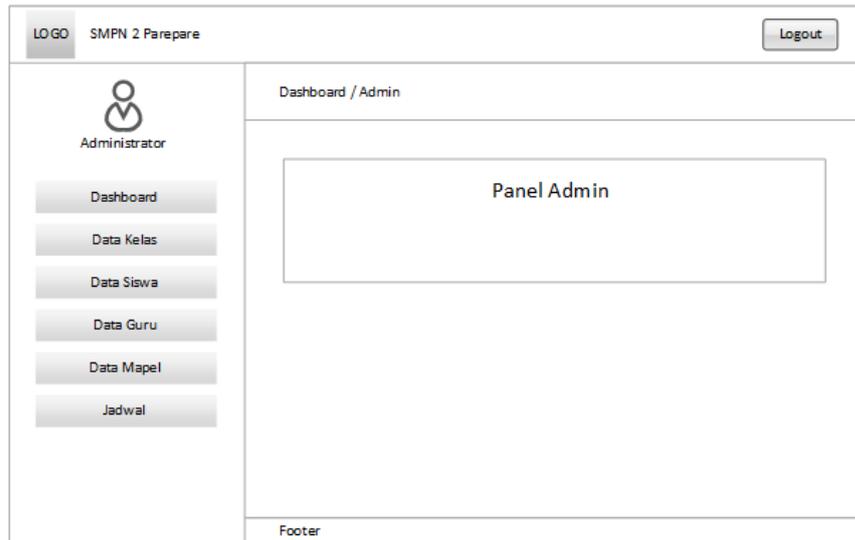
#### 2. Tampilan *Login Siswa (Android)*



The image shows an Android login interface. At the top center is the text "LOGO". Below this are two input fields: the first is labeled "Email" and the second is labeled "Password". At the bottom center of the form is a button labeled "Login".

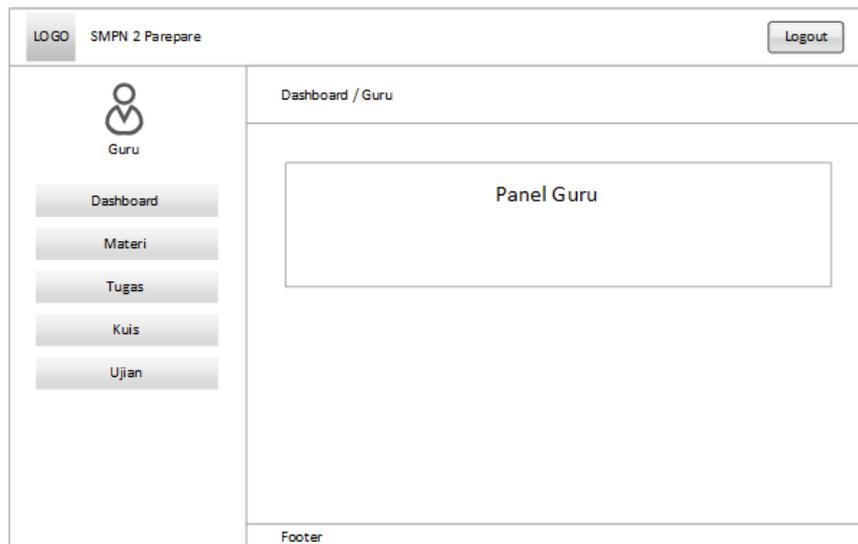
**Gambar III.29** *Interface Login Siswa*

### 3. Tampilan *Web Admin*



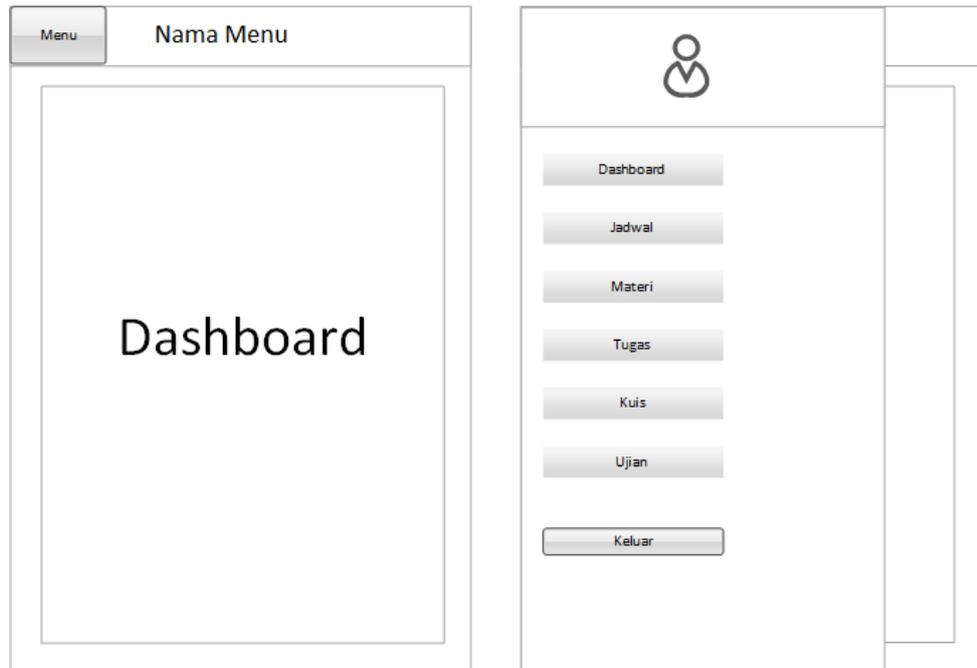
**Gambar III.30** *Interface Web Admin*

### 4. Tampilan *Web Guru*



**Gambar III.31** *Interface Web Guru*

## 5. Tampilan Android Siswa



**Gambar III.32** *Interface* Aplikasi Android Siswa

## BAB IV

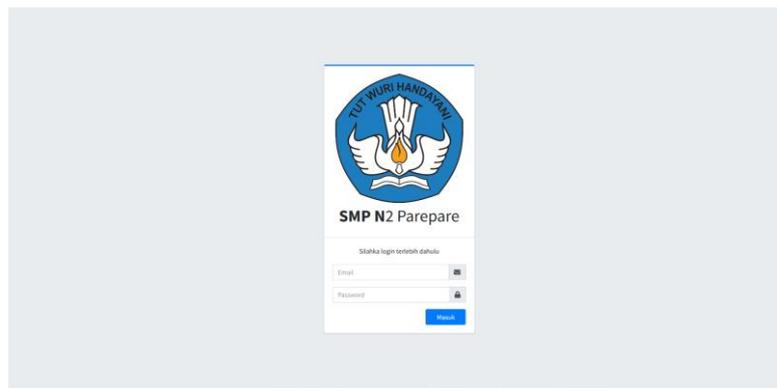
### HASIL DAN PEMBAHASAN

#### IV.1 Hasil

Berdasarkan dari penelitian yang telah dilakukan, maka hasil yang didapatkan dari Aplikasi E-Learning Berbasis Android pada SMP Negeri 2 Parepare adalah sebagai berikut.

##### IV.1.1 Tampilan Admin

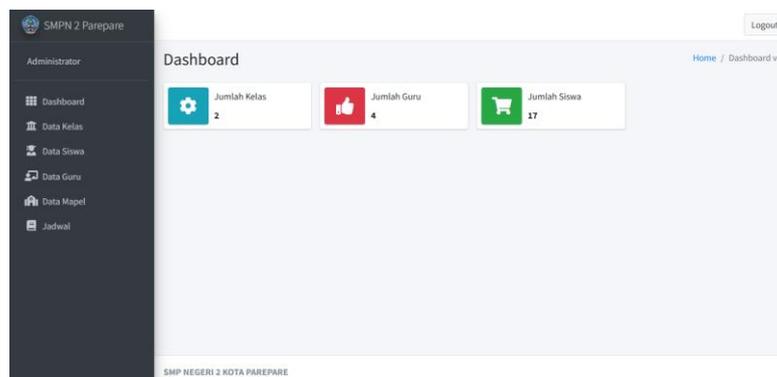
###### 1. Tampilan *Login* Admin



**Gambar IV.1** Tampilan *Login* Admin

Gambar diatas menunjukkan halaman *login* admin (begitu pula dengan guru) pada *website e-learning*. Untuk mengakses dan mengelola data dalam aplikasi *e-learning*, admin (sekaligus guru) harus *login* terlebih dahulu menggunakan alamat *email* dan *password* yang telah terdaftar di *database*.

###### 2. Tampilan *Dashboard* Admin



**Gambar IV.2** Tampilan *Dashboard* Admin

Gambar diatas menunjukkan halaman *dashboard* admin, yang merupakan ringkasan dari jumlah data dari beberapa menu admin.

### 3. Tampilan Data Kelas

NO.	Nama Kelas	Tahun Ajaran	Semester	Wali Kelas	Aksi
1	7E	2023	ganjil	Gustinah Laewang, S. Pd	Aksi
2	8B	2023	ganjil	Rahmaniar, S. Pd	Aksi

**Gambar IV.3** Tampilan Data Kelas

Halaman tampilan admin untuk data kelas terlihat pada gambar di atas.

### 4. Tampilan Data Siswa

NO.	NIS	Nama	Kelas	Agama	Alamat	Jenis Kelamin	Aksi
1	202326001	Andi Zahrah Athira	7E	Islam	Parepare	Perempuan	Aksi
2	202326002	Hafidzhah Umaimah	7E	Islam	Parepare	Perempuan	Aksi
3	202326003	Afia Dzil Athifa	7E	Islam	Parepare	Laki-laki	Aksi
4	202326004	Afwa Dzul Agilah	7E	Islam	Parepare	Perempuan	Aksi
5	202326005	Alia Asnita	7E	Islam	Parepare	Laki-laki	Aksi
6	202326006	Alya Amani Amandini	7E	Islam	Parepare	Perempuan	Aksi

**Gambar IV.4** Tampilan Data Siswa

Halaman tampilan data siswa, tempat admin dapat mengelola data siswa, terlihat pada gambar di atas.

## 5. Tampilan Data Guru

Dashboard

Data Guru

Copy CSV Excel PDF Print Column visibility Search:

NO.	NIP	Nama	Jenis Kelamin	Aktif	Wali	Aksi
1	198807272019032004	Gustinah Laewang, S. Pd	Perempuan	Y	Y	Aksi
2	197510292005022003	Masnah, S. Pd	Perempuan	Y	N	Aksi
3	196809141994122001	Rahmaniar, S. Pd	Perempuan	Y	Y	Aksi
4	198012102007012014	Darmawati, S. Pd	Perempuan	Y	N	Aksi

Showing 1 to 4 of 4 entries Previous 1 Next

**Gambar IV.5** Tampilan Data Guru

Halaman tampilan data guru, tempat admin dapat mengelola data guru, terlihat pada gambar di atas.

## 6. Tampilan Data Mata Pelajaran

Dashboard

Data Kelas

Copy CSV Excel PDF Print Column visibility Search:

NO.	Nama Mata Pelajaran	Guru	Aksi
1	Bahasa Inggris	Rahmaniar, S. Pd	Aksi
2	IPS	Masnah, S. Pd	Aksi
3	Seni dan Prakarya	Rahmaniar, S. Pd	Aksi
4	PJOK	Darmawati, S. Pd	Aksi
5	Pkn	Darmawati, S. Pd	Aksi
6	Matematika	Gustinah Laewang, S. Pd	Aksi

**Gambar IV.6** Tampilan Data Mapel

Gambar diatas menunjukkan halaman tampilan data mata pelajaran dimana admin dapat mengelola data mata pelajaran.

## 7. Tampilan Data Jadwal

NO.	Nama Kelas	Tahun Ajaran	Semester	Wali Kelas	Jadwal
1	7E	2023	ganjil	Gustinah Laewang, S. Pd	Jadwal
2	8B	2023	ganjil	Rahmaniar, S. Pd	Jadwal

**Gambar IV.7** Tampilan Data Jadwal

Gambar diatas menunjukkan halaman tampilan data jadwal dimana admin dapat mengelola data jadwal.

### IV.1.2 Tampilan Guru

#### 1. Tampilan *Dashboard* Guru

Menu	Jumlah
Kelas Mengajar	1
Mata Pelajaran Men...	4
Materi Mata Pelajaran	2

**Gambar IV.8** Tampilan *Dashboard* Guru

Gambar diatas menunjukkan halaman *dashboard* guru, yang merupakan ringkasan dari jumlah data dari beberapa menu guru.

## 2. Tampilan Materi Guru

Dashboard

Materi Pelajaran Matematika

Copy CSV Excel PDF Print Column visibility Search:

NO.	Nama Mata Pelajaran	Pertemuan Ke-	Judul	Tingkat	Tahun Ajaran	Deskripsi	File	Aksi
1	Matematika	Ke-1	Bilangan	7	2023	Bab 1 Bilangan		<a href="#">Edit</a> <a href="#">Hapus</a>
2	Matematika	Ke-2	Himpunan	7	2023	Bab 2 Himpunan		<a href="#">Edit</a> <a href="#">Hapus</a>

Showing 1 to 2 of 2 entries

Previous 1 Next

**Gambar IV.9** Tampilan Kelola Materi

Gambar diatas menunjukkan halaman tampilan data materi dimana guru dapat mengelola data materi.

## 3. Tampilan Tugas Guru

Dashboard

Data Tugas Kelas

Copy CSV Excel PDF Print Column visibility Search:

NO.	Nama Siswa	Tugas Ke-	Lihat Tugas	Nilai
1	Andi Zahrah Athira	1	<a href="#">Buka</a>	85
2	Hafidzhah Umairah	1		
3	Afia Dzil Athifa	1		
4	Afwa Dzul Agilah	1		
5	Alla Asnita	1		
6	Alya Amani Amandini	1		
7	April Hidayanti	1		

**Gambar IV.10** Tampilan Kelola Tugas

Gambar diatas menunjukkan halaman tampilan data tugas dimana guru dapat mengelola data tugas.

#### 4. Tampilan Kuis Guru

NO.	Nama Siswa	Kuis Ke-	Lihat Kuis	Nilai
1	Andi Zahrah Athira	1	Buka	73
2	Hafidzhah Umaimah	1		
3	Afia Dzil Athifa	1		
4	Afwa Dzul Agilah	1		
5	Alia Asnita	1		
6	Alya Amani Amandini	1		
7	April Hidayanti	1		

**Gambar IV.11** Tampilan Kelola Kuis

Gambar diatas menunjukkan halaman tampilan data kuis dimana guru dapat mengelola data kuis.

#### 5. Tampilan Ujian Guru

NO.	Mapel	Ujian Ke-	Jumlah Soal Tersedia	Mulai Ujian
1	Matematika	1	16 & 5	Nilai Soal

**Gambar IV.12** Tampilan Kelola Ujian

Gambar diatas menunjukkan halaman tampilan data ujian dimana guru dapat mengelola data ujian.

### IV.1.3 Tampilan Siswa

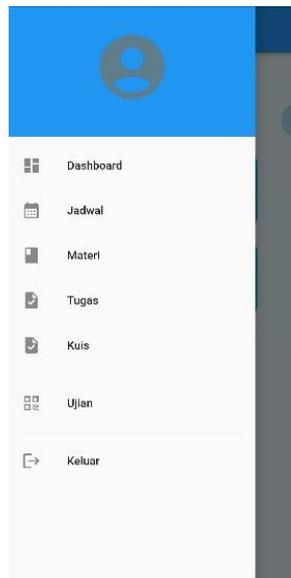
#### 1. Tampilan *Login* Siswa



**Gambar IV.13** Tampilan *Login* Siswa

Gambar diatas adalah tampilan saat aplikasi pertama dijalankan apabila *user* belum melakukan *login*.

#### 2. Tampilan Menu Siswa



**Gambar IV.14** Tampilan Menu Siswa

Gambar diatas adalah tampilan menu-menu yang ada pada aplikasi.

### 3. Tampilan *Dashboard* Siswa



**Gambar IV.15** Tampilan *Dashboard* Siswa

Gambar diatas adalah tampilan aplikasi saat siswa telah melakukan *login*, yang merupakan ringkasan dari jumlah data dari beberapa menu siswa.

### 4. Tampilan Jadwal Siswa

The screenshot shows a mobile application interface for a student's schedule. At the top, there is a blue header with a hamburger menu icon and the word "Jadwal". Below the header, the title "Jadwal Pelajaran" is centered. A table with three columns and four rows is displayed. The columns are labeled "Hari", "Mapel", and "Jam Pelajaran". The rows show the schedule for Monday (Senin) with subjects Pjok, Matematika, and IPA at different times.

Hari	Mapel	Jam Pelajaran
Senin	PJOK	07:00:00
Senin	Matematika	07:30:00
Senin	IPA	09:50:00

**Gambar IV.16** Tampilan Jadwal Siswa

Gambar diatas adalah tampilan jadwal yang diatur untuk siswa.

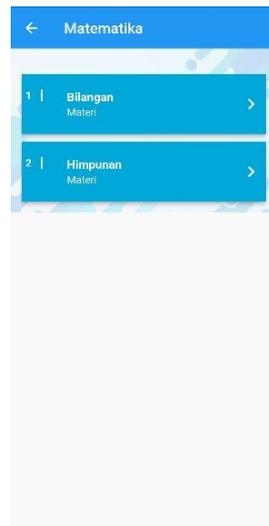
## 5. Tampilan Daftar Mata Pelajaran Siswa



**Gambar IV.17** Tampilan Daftar Mata Pelajaran

Gambar diatas adalah tampilan setelah siswa memilih menu materi, tugas, atau kuis. Maka akan tampil tampilan *list* mata pelajaran yang ada berdasarkan yang diatur oleh admin.

## 6. Tampilan Materi Siswa



**Gambar IV.18** Tampilan Materi Siswa

Gambar diatas adalah tampilan materi siswa berdasarkan mata pelajaran yang diprogramkan oleh siswa.

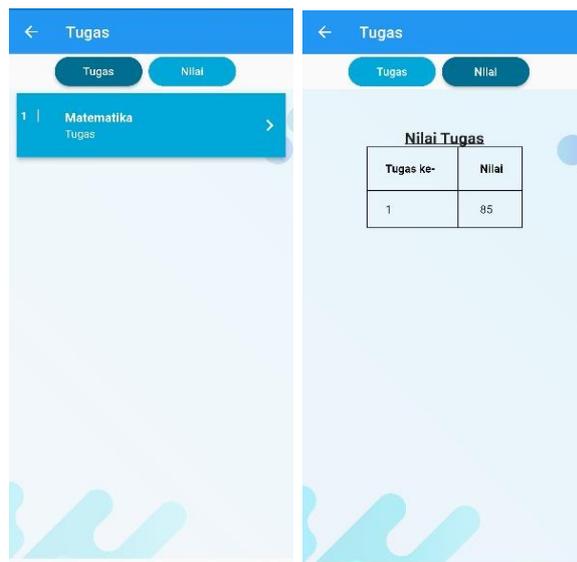
## 7. Tampilan Detail Materi



**Gambar IV.19** Tampilan Detail Materi

Gambar diatas adalah contoh tampilan dari materi pdf yang tersedia pada mata pelajaran. Beserta dengan tombol tugas, yang mengarah ke tugas materi tersebut.

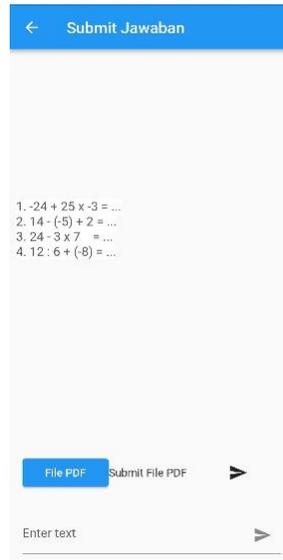
## 8. Tampilan Tugas Siswa



**Gambar IV.20** Tampilan Tugas Siswa

Gambar diatas merupakan tampilan *list* tugas yang ada pada mata pelajaran.

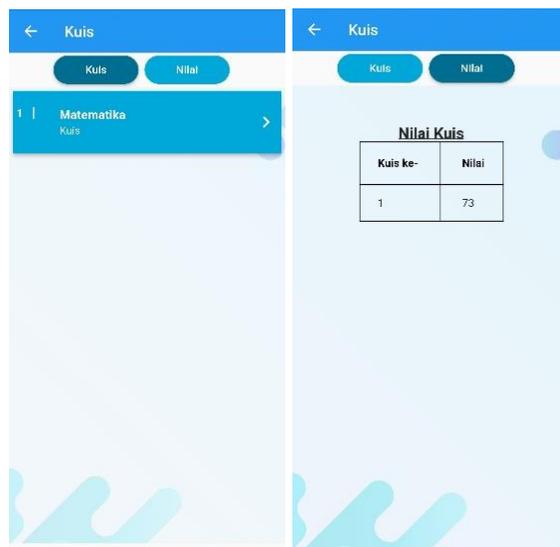
## 9. Tampilan Submit Tugas



**Gambar IV.21** Tampilan *Submit* Tugas

Gambar diatas merupakan tampilan soal dari tugas yang dipilih sebelumnya. Setelah memilih *list* tugas, jawaban dapat di-*submit* dengan mengupload *file* pdf atau mengetik manual jawaban pada *form* yang tersedia.

## 10. Tampilan Kuis Siswa



**Gambar IV.22** Tampilan Kuis Siswa

Gambar diatas merupakan tampilan *list* kuis yang ada pada mata pelajaran.

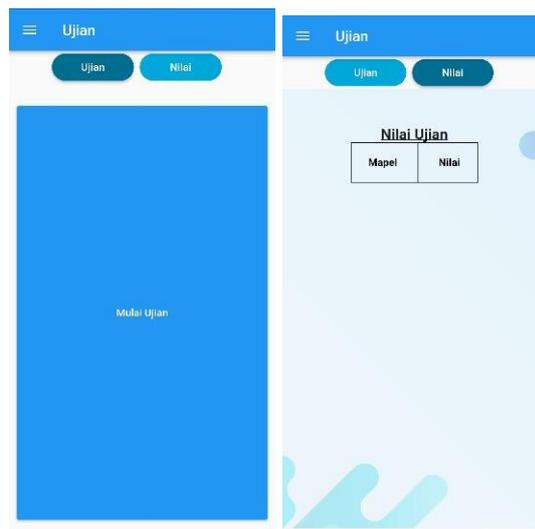
## 11. Tampilan Submit Kuis



**Gambar IV.23** Tampilan *Submit* Kuis

Gambar diatas merupakan tampilan soal dari kuis yang dipilih sebelumnya. Setelah memilih list kuis, jawaban dapat di-*submit* dengan mengupload *file* pdf atau mengetik manual jawaban pada *form* yang tersedia.

## 12. Tampilan Ujian Siswa

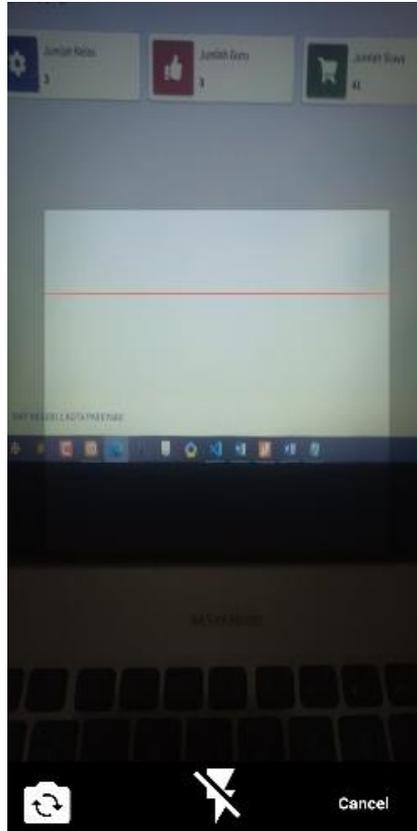


**Gambar IV.24** Tampilan Ujian Siswa

Gambar diatas merupakan tampilan yang akan tampil pada saat siswa memilih ujian. Tampilan ujian dapat dilihat pada gambar IV.26, dimana soal

ujian ada yang berbentuk pilihan ganda dan berbentuk essay. Sistem penilaian ujian yaitu 60% nilai dari soal pilihan ganda dan 40% nilai dari soal essay, dan nantinya akan di total dan mendapatkan total nilai ujian siswa.

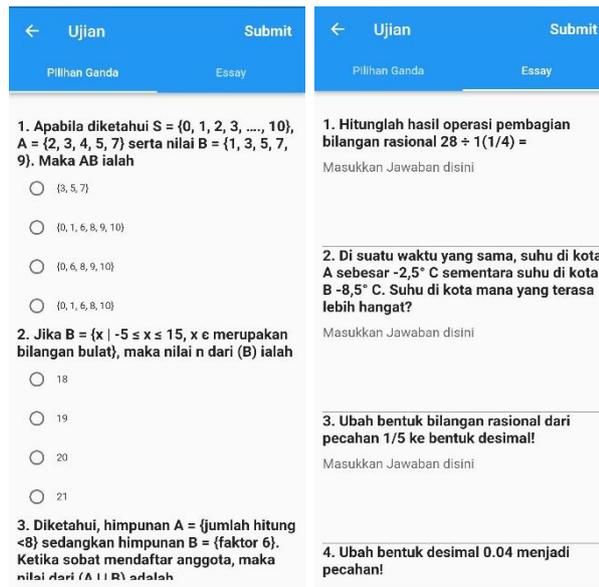
### 13. Tampilan *Scan QR*



**Gambar IV.25** Tampilan *Scan QR*

Gambar diatas merupakan tampilan setelah siswa menekan tombol *scan qr*. Setelah kamera diarahkan kepada qr yang ada maka akan tampil soal yang sudah disiapkan sebelumnya oleh guru.

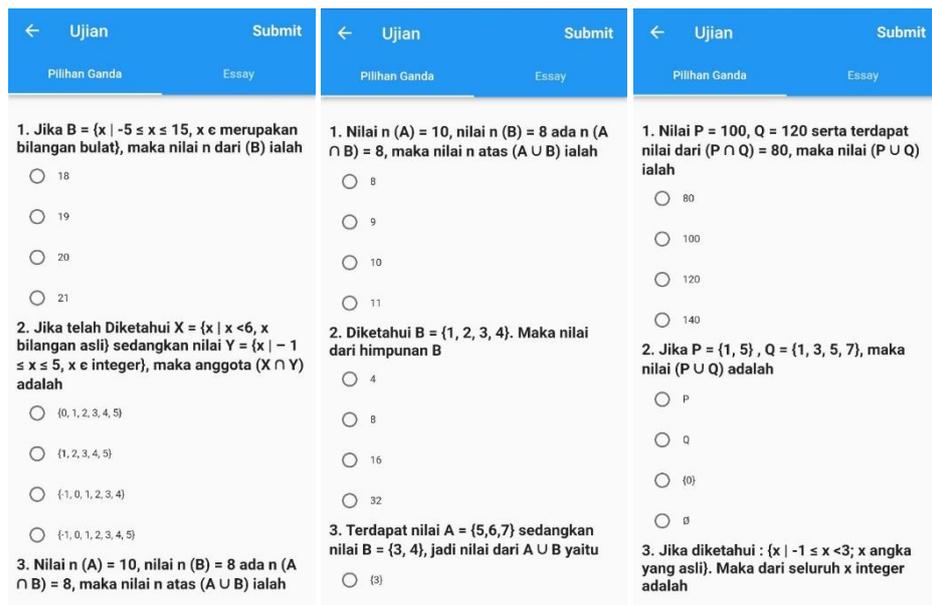
## 14. Tampilan Mengerjakan Soal Ujian



Gambar IV.26 Tampilan Soal Ujian

Gambar diatas merupakan tampilan saat *scan* qr menemukan soal yang akan dikerjakan.

## 15. Tampilan Pengacakan Soal pada Pilihan Ganda



Gambar IV.27 Pengacakan Soal Pilihan Ganda

Gambar diatas merupakan contoh tampilan pengacakan soal pada soal pilihan ganda.

## IV.2 Pembahasan

### IV.2.1 Pengujian *Black Box*

*Black box testing* merupakan pengujian perangkat lunak dari sisi fungsional tanpa memperhatikan sisi desain dan kode pada program. Pengujian dilakukan untuk mengetahui apakah fungsi masukan (*input*) dan fungsi keluaran (*output*) dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

#### 1. Pengujian *Website Admin*

**Tabel IV.1** Instrumen Pengujian *Black Box* pada *Website Admin*

Kasus dan Hasil Uji				
Komponen yang diuji	Data yang dimasukkan	Yang Diharapkan	Pengamatan	Hasil
Halaman Login Admin	<i>Input</i> email	Melakukan validasi <i>inputan</i>	Berhasil melakukan validasi	Sesuai
	<i>Input</i> password	Melakukan validasi <i>inputan</i>	Berhasil melakukan validasi	Sesuai
	Klik login	Mengarahkan admin ke halaman dashboard	Berhasil mengarahkan admin ke halaman dashboard	Sesuai
Menu Admin	Menu Data Kelas	Menampilkan halaman data kelas	Berhasil Menampilkan halaman data kelas	Sesuai
	Menu Data Siswa	Menampilkan halaman data siswa	Berhasil menampilkan halaman Data siswa	Sesuai
	Menu Data Guru	Menampilkan halaman data guru	Berhasil menampilkan halaman data guru	Sesuai
	Menu Data Mapel	Menampilkan halaman data mapel	Berhasil Menampilkan halaman data mapel	Sesuai
	Menu Jadwal	Menampilkan halaman data jadwal	Berhasil menampilkan halaman data jadwal	Sesuai

<b>Kasus dan Hasil Uji</b>				
<b>Komponen yang diuji</b>	<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Halaman Data Kelas Admin	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkanform edit data	Sesuai
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai
Halaman Data Siswa Admin	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkanform edit data	Sesuai
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai
Halaman Data Guru Admin	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkan form edit data	Sesuai
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai
Halaman Data Mapel Admin	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkan form edit data	Sesuai

Kasus dan Hasil Uji				
Komponen yang diuji	Data yang dimasukkan	Yang Diharapkan	Pengamatan	Hasil
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai
Halaman Jadwal Admin	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkan form edit data	Sesuai
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai

## 2. Pengujian *Website* Guru

**Tabel IV.2** Instrumen Pengujian *Black Box* pada *Website* Guru

Kasus dan Hasil Uji				
Komponen yang diuji	Data yang dimasukkan	Yang Diharapkan	Pengamatan	Hasil
Menu Guru	Menu Data Materi	Menampilkan halaman data materi	Berhasil Menampilkan halaman data materi	Sesuai
	Menu Data Tugas	Menampilkan halaman data tugas	Berhasil menampilkan halaman Data tugas	Sesuai
	Menu Data Kuis	Menampilkan halaman data kuis	Berhasil menampilkan halaman data kuis	Sesuai
	Menu Data Ujian	Menampilkan halaman data ujian	Berhasil Menampilkan halaman data ujian	Sesuai

<b>Kasus dan Hasil Uji</b>				
<b>Komponen yang diuji</b>	<b>Data yang dimasukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Hasil</b>
Halaman Materi Guru	Tombol Detail Mata Pelajaran	Menampilkan data detail mata pelajaran	Berhasil Menampilkan data detail mata pelajaran	Sesuai
Halaman Detail Materi Guru	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkan form edit data	Sesuai
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai
Halaman Tugas Guru	Tombol Tugas Mata Pelajaran	Menampilkan data detail Tugas mata pelajaran	Berhasil Menampilkan data detail tugas mata pelajaran	Sesuai
Halaman Detail Tugas Guru	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkan form edit data	Sesuai
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai
Halaman Kuis Guru	Tombol Kuis Mata Pelajaran	Menampilkan data detail Kuis mata pelajaran	Berhasil Menampilkan data detail Kuis mata pelajaran	Sesuai
Halaman Detail Kuis Guru	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Edit	Menampilkan form edit data	Berhasil menampilkan form edit data	Sesuai

Kasus dan Hasil Uji				
Komponen yang diuji	Data yang dimasukkan	Yang Diharapkan	Pengamatan	Hasil
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai
Halaman Ujian Guru	Tombol Ujian Mata Pelajaran	Menampilkan data detail ujian mata pelajaran	Berhasil Menampilkan data detail ujian mata pelajaran	Sesuai
Halaman Detail Ujian Guru	Tombol Tambah Data	Menampilkan form tambah data	Berhasil Menampilkan form tambah data	Sesuai
	Tombol Aksi Soal	Menampilkan form data bank soal	Berhasil menampilkan data bank soal	Sesuai
	Tombol Aksi Hapus	Menampilkan konfirmasi hapus data	Berhasil menampilkan Konfirmasi hapus data	Sesuai

### 3. Pengujian Aplikasi Android Siswa

**Tabel IV.3** Instrumen Pengujian *Black Box* pada Android Siswa

Kasus dan Hasil Uji				
Komponen yang diuji	Data yang dimasukkan	Yang Diharapkan	Pengamatan	Hasil
Tampilan Login Android	<i>Input</i> email	Melakukan validasi <i>inputan</i>	Berhasil melakukan validasi	Sesuai
	<i>Input</i> password	Melakukan validasi <i>inputan</i>	Berhasil melakukan validasi	Sesuai
	Klik login	Mengarahkan ke halaman utama android	Berhasil mengarahkan ke halaman utama android	Sesuai
Tampilan Utama Android	Menu Data Materi	Menampilkan halaman data materi	Berhasil Menampilkan halaman data materi	Sesuai

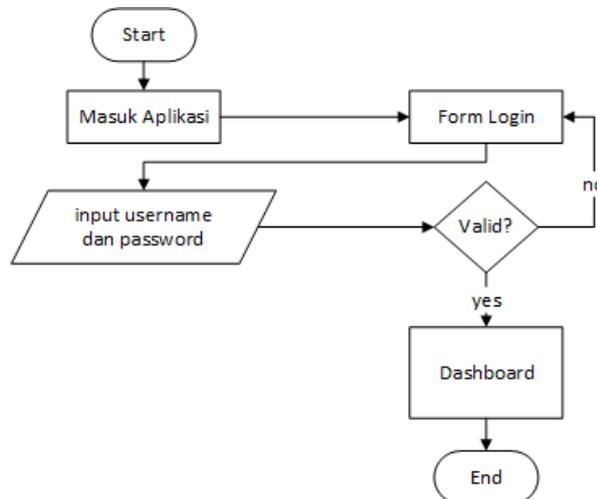
Kasus dan Hasil Uji				
Komponen yang diuji	Data yang dimasukkan	Yang Diharapkan	Pengamatan	Hasil
	Menu Data Tugas	Menampilkan halaman data tugas	Berhasil menampilkan halaman Data tugas	Sesuai
	Menu Data Kuis	Menampilkan halaman data kuis	Berhasil menampilkan Dalaman data kuis	Sesuai
	Menu Data Ujian	Menampilkan halaman scan QR ujian	Berhasil Menampilkan halaman scan QR ujian	Sesuai

#### IV.2.2 Pengujian *White Box*

*White box testing* merupakan teknik menguji aplikasi dengan menganalisa kode program pada sisi logika apakah program memiliki kesalahan atau tidak. Jika kode telah di produksi ke dalam output dan tidak memenuhi persyaratan maka kode akan dikompilasi hingga mencapai hasil yang diharapkan. Pada pengujian *whitebox* akan digunakan teknik *basis path*.

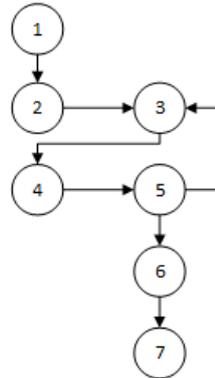
##### 1. Pengujian *White Box* pada *Login*

###### a. *Flowchart*



**Gambar IV.28** *Flowchart Login*

b. *Flowgraph*



**Gambar IV.29** *Flowgraph Login*

Berdasarkan *flowgraph login* diatas terdapat 7 *edge*, 7 *node*, dan 2 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

E = *Edge*

N = *Node*

Maka, persamaan dari rumus diatas adalah  $V(G) = 7 - 7 + 2 = 2$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 2$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

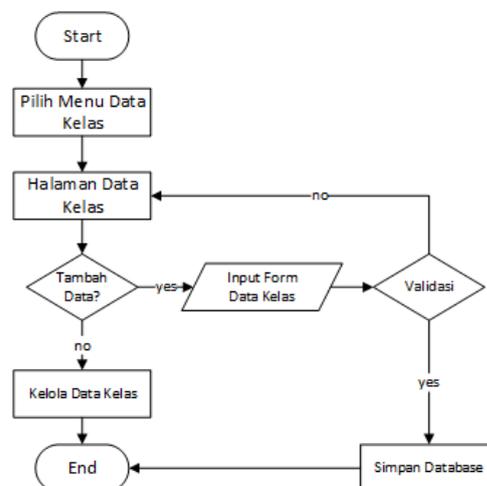
**Tabel IV.4** *Test Case Login*

<i>Path</i>	1
Jalur	1-2-3-4-5-6-7
Skenario	1. <i>Start</i> 2. <i>Masuk aplikasi</i> 3. <i>Form login</i> 4. <i>Input username dan password</i> 5. <i>Cek validasi username dan password sesuai</i> 6. <i>Menampilkan dashboard</i>

	7. <i>End</i>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-5-3-4-5-6-7
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Masuk aplikasi</li> <li>3. <i>Form login</i></li> <li>4. <i>Input username dan password</i></li> <li>5. Cek validasi <i>username dan password</i> tidak sesuai</li> <li>6. <i>Form login</i></li> <li>7. <i>Input username dan password</i></li> <li>8. Cek validasi <i>username dan password</i> sesuai</li> <li>9. Menampilkan <i>dashboard</i></li> <li>10. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

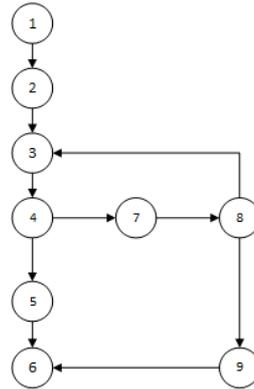
2. Pengujian *White Box* pada Data Kelas oleh Admin

a. *Flowchart*



**Gambar IV.30** *Flowchart* Data Kelas oleh Admin

b. *Flowgraph*



**Gambar IV.31** *Flowgraph* Data Kelas oleh Admin

Berdasarkan *flowgraph* data kelas oleh admin diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 10 - 9 + 2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

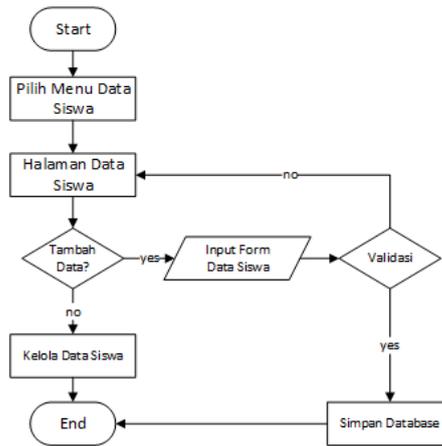
**Tabel IV.5** *Test Case* Data Kelas oleh Admin

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	1. <i>Start</i> 2. Pilih menu data kelas 3. Halaman data kelas 4. Tambah data tidak dilakukan 5. Kelola data kelas 6. <i>End</i>
Hasil Pengujian	Berhasil

<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data kelas</li> <li>3. Halaman data kelas</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data kelas</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> <li>8. <i>End</i></li> </ol>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data kelas</li> <li>3. Halaman data kelas</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data kelas</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data kelas</li> <li>8. Tambah data dilakukan</li> <li>9. <i>Input form</i> data kelas</li> <li>10. Cek validasi sesuai</li> <li>11. Data tersimpan di <i>database</i></li> <li>12. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

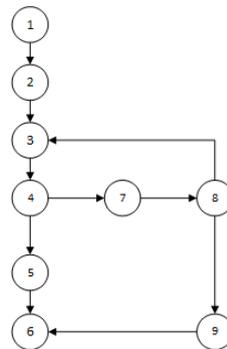
### 3. Pengujian *White Box* pada Data Siswa oleh Admin

#### a. *Flowchart*



**Gambar IV.32** *Flowchart* Data Siswa oleh Admin

#### b. *Flowgraph*



**Gambar IV.33** *Flowgraph* Data Siswa oleh Admin

Berdasarkan *flowgraph* data siswa oleh admin diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

E = *Edge*

N = *Node*

Maka, persamaan dari rumus diatas adalah  $V(G) = 10 - 9 + 2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

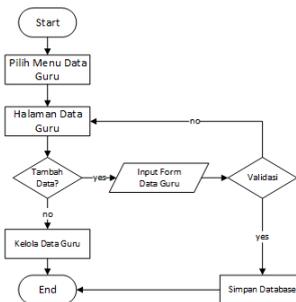
**Tabel IV.6** *Test Case* Data Siswa oleh Admin

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data siswa</li> <li>3. Halaman data siswa</li> <li>4. Tambah data tidak dilakukan</li> <li>5. Kelola data siswa</li> <li>6. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data siswa</li> <li>3. Halaman data siswa</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data siswa</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> <li>8. <i>End</i></li> </ol>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data siswa</li> <li>3. Halaman data siswa</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data siswa</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data siswa</li> </ol>

	8. Tambah data dilakukan 9. <i>Input form</i> data siswa 10. Cek validasi sesuai 11. Data tersimpan di <i>database</i> 12. <i>End</i>
Hasil pengujian	Berhasil

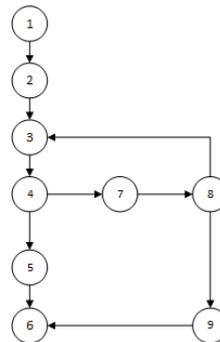
4. Pengujian *White Box* pada Data Guru oleh Admin

a. *Flowchart*



**Gambar IV.34** *Flowchart* Data Guru oleh Admin

b. *Flowgraph*



**Gambar IV.35** *Flowgraph* Data Guru oleh Admin

Berdasarkan *flowgraph* data guru oleh admin diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

E = *Edge*

N = *Node*

Maka, persamaan dari rumus diatas adalah  $V(G) = 10-9+2 = 3$ , atau dapat menggunakan rumus  $V(G) = region$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

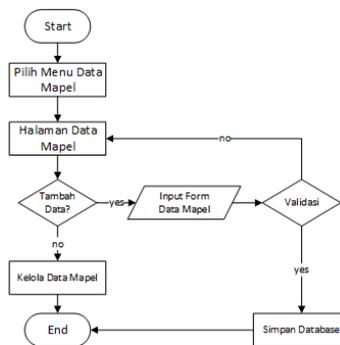
**Tabel IV.7** Test Case Data Guru oleh Admin

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data guru</li> <li>3. Halaman data guru</li> <li>4. Tambah data tidak dilakukan</li> <li>5. Kelola data guru</li> <li>6. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data guru</li> <li>3. Halaman data guru</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data guru</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> <li>8. <i>End</i></li> </ol>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data guru</li> <li>3. Halaman data guru</li> </ol>

	<ol style="list-style-type: none"> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data guru</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data guru</li> <li>8. Tambah data dilakukan</li> <li>9. <i>Input form</i> data guru</li> <li>10. Cek validasi sesuai</li> <li>11. Data tersimpan di <i>database</i></li> <li>12. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

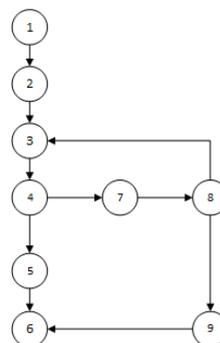
5. Pengujian *White Box* pada Data Mapel oleh Admin

a. *Flowchart*



**Gambar IV.36** *Flowchart* Data Mapel oleh Admin

b. *Flowgraph*



**Gambar IV.37** *Flowgraph* Data Mapel oleh Admin

Berdasarkan *flowgraph* data mapel oleh admin diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 10 - 9 + 2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

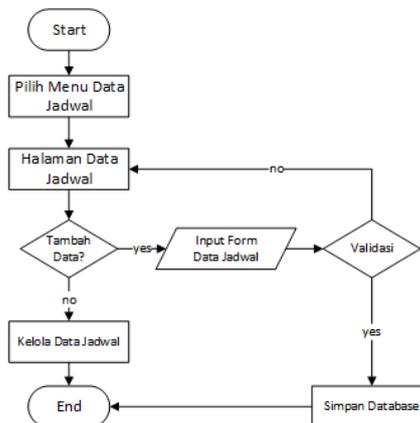
**Tabel IV.8** *Test Case* Data Mapel oleh Admin

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data mapel</li> <li>3. Halaman data mapel</li> <li>4. Tambah data tidak dilakukan</li> <li>5. Kelola data mapel</li> <li>6. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data mapel</li> <li>3. Halaman data mapel</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data mapel</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> </ol>

	8. <i>End</i>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data mapel</li> <li>3. Halaman data mapel</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data mapel</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data mapel</li> <li>8. Tambah data dilakukan</li> <li>9. <i>Input form</i> data mapel</li> <li>10. Cek validasi sesuai</li> <li>11. Data tersimpan di <i>database</i></li> <li>12. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

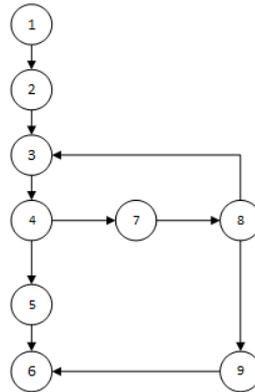
6. Pengujian *White Box* pada Data Jadwal oleh Admin

a. *Flowchart*



**Gambar IV.38** *Flowchart* Data Jadwal oleh Admin

b. *Flowgraph*



**Gambar IV.39** *Flowgraph* Data Jadwal oleh Admin

Berdasarkan *flowgraph* data jadwal oleh admin diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

E = *Edge*

N = *Node*

Maka, persamaan dari rumus diatas adalah  $V(G) = 10 - 9 + 2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

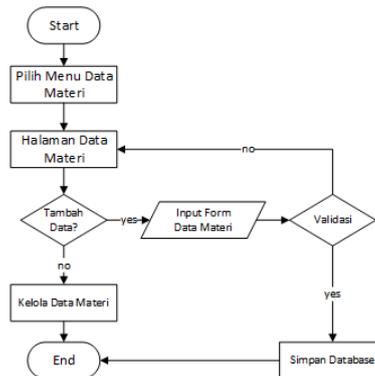
**Tabel IV.9** *Test Case* Data Jadwal oleh Admin

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	1. <i>Start</i> 2. Pilih menu data jadwal 3. Halaman data jadwal 4. Tambah data tidak dilakukan 5. Kelola data jadwal 6. <i>End</i>

Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data jadwal</li> <li>3. Halaman data jadwal</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data jadwal</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> <li>8. <i>End</i></li> </ol>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data jadwal</li> <li>3. Halaman data jadwal</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data jadwal</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data jadwal</li> <li>8. Tambah data dilakukan</li> <li>9. <i>Input form</i> data jadwal</li> <li>10. Cek validasi sesuai</li> <li>11. Data tersimpan di <i>database</i></li> <li>12. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

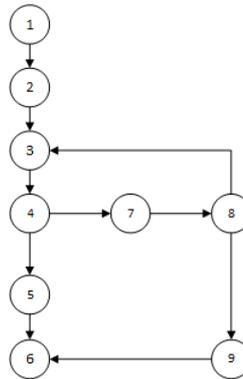
## 7. Pengujian *White Box* pada Data Materi oleh Guru

### a. *Flowchart*



**Gambar IV.40** *Flowchart* Data Materi oleh Guru

### b. *Flowgraph*



**Gambar IV.41** *Flowgraph* Data Materi oleh Guru

Berdasarkan *flowgraph* data materi oleh guru diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 10 - 9 + 2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

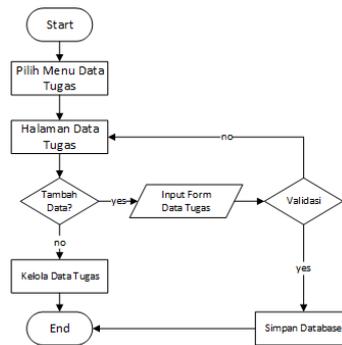
**Tabel IV.10** *Test Case Data Materi* oleh Guru

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data materi</li> <li>3. Halaman data materi</li> <li>4. Tambah data tidak dilakukan</li> <li>5. Kelola data materi</li> <li>6. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data materi</li> <li>3. Halaman data materi</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data materi</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> <li>8. <i>End</i></li> </ol>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data materi</li> <li>3. Halaman data materi</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data materi</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data materi</li> </ol>

	8. Tambah data dilakukan 9. <i>Input form</i> data materi 10. Cek validasi sesuai 11. Data tersimpan di <i>database</i> 12. <i>End</i>
Hasil pengujian	Berhasil

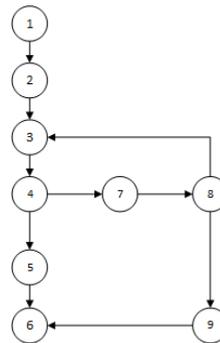
8. Pengujian *White Box* pada Data Tugas oleh Guru

a. *Flowchart*



**Gambar IV.42** *Flowchart* Data Tugas oleh Guru

b. *Flowgraph*



**Gambar IV.43** *Flowgraph* Data Tugas oleh Guru

Berdasarkan *flowgraph* data tugas oleh guru diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 10-9+2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

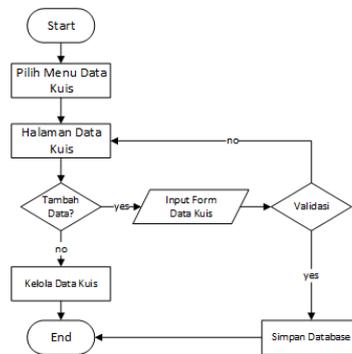
**Tabel IV.11** *Test Case* Data Tugas oleh Guru

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data tugas</li> <li>3. Halaman data tugas</li> <li>4. Tambah data tidak dilakukan</li> <li>5. Kelola data tugas</li> <li>6. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data tugas</li> <li>3. Halaman data tugas</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data tugas</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> <li>8. <i>End</i></li> </ol>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data tugas</li> </ol>

	<ol style="list-style-type: none"> <li>3. Halaman data tugas</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data tugas</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data tugas</li> <li>8. Tambah data dilakukan</li> <li>9. <i>Input form</i> data tugas</li> <li>10. Cek validasi sesuai</li> <li>11. Data tersimpan di <i>database</i></li> <li>12. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

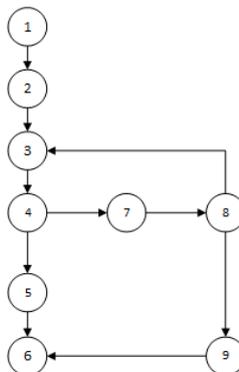
9. Pengujian *White Box* pada Data Kuis oleh Guru

a. *Flowchart*



**Gambar IV.44** *Flowchart* Data Kuis oleh Guru

b. *Flowgraph*



**Gambar IV.45** *Flowgraph* Data Kuis oleh Guru

Berdasarkan *flowgraph* data kuis oleh guru diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 10 - 9 + 2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

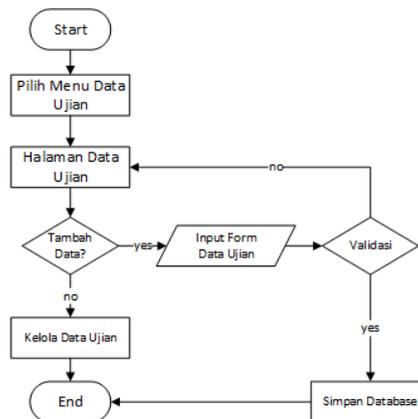
**Tabel IV.12** *Test Case* Data Kuis oleh Guru

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data kuis</li> <li>3. Halaman data kuis</li> <li>4. Tambah data tidak dilakukan</li> <li>5. Kelola data kuis</li> <li>6. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data kuis</li> <li>3. Halaman data kuis</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data kuis</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> </ol>

	8. <i>End</i>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data kuis</li> <li>3. Halaman data kuis</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data kuis</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data kuis</li> <li>8. Tambah data dilakukan</li> <li>9. <i>Input form</i> data kuis</li> <li>10. Cek validasi sesuai</li> <li>11. Data tersimpan di <i>database</i></li> <li>12. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

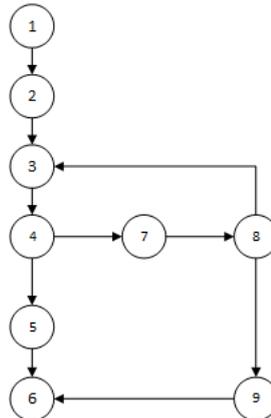
## 10. Pengujian *White Box* pada Data Ujian oleh Guru

### a. *Flowchart*



**Gambar IV.46** *Flowchart* Data Ujian oleh Guru

b. *Flowgraph*



**Gambar IV.47** *Flowgraph* Data Ujian oleh Guru

Berdasarkan *flowgraph* data ujian oleh guru diatas terdapat 10 *edge*, 9 *node*, dan 3 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

E = *Edge*

N = *Node*

Maka, persamaan dari rumus diatas adalah  $V(G) = 10 - 9 + 2 = 3$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 3$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

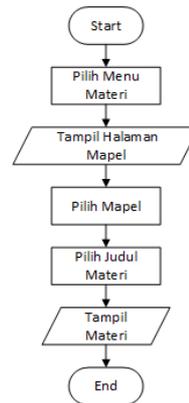
**Tabel IV.13** *Test Case* Data Ujian oleh Guru

<i>Path</i>	1
Jalur	1-2-3-4-5-6
Skenario	1. <i>Start</i> 2. Pilih menu data ujian 3. Halaman data ujian 4. Tambah data tidak dilakukan 5. Kelola data ujian

	6. <i>End</i>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data ujian</li> <li>3. Halaman data ujian</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data ujian</li> <li>6. Cek validasi sesuai</li> <li>7. Data tersimpan di <i>database</i></li> <li>8. <i>End</i></li> </ol>
Hasil pengujian	Berhasil
<i>Path</i>	3
Jalur	1-2-3-4-7-8-3-4-7-8-9-6
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu data ujian</li> <li>3. Halaman data ujian</li> <li>4. Tambah data dilakukan</li> <li>5. <i>Input form</i> data ujian</li> <li>6. Cek validasi tidak sesuai</li> <li>7. Halaman data ujian</li> <li>8. Tambah data dilakukan</li> <li>9. <i>Input form</i> data ujian</li> <li>10. Cek validasi sesuai</li> <li>11. Data tersimpan di <i>database</i></li> <li>12. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

## 11. Pengujian *White Box* pada Akses Data Materi oleh Siswa

### a. *Flowchart*



**Gambar IV.48** *Flowchart* Akses Materi oleh Siswa

### b. *Flowgraph*



**Gambar IV.49** *Flowgraph* Akses Materi oleh Siswa

Berdasarkan *flowgraph* akses materi oleh siswa diatas terdapat 6 *edge*, 7 *node*, dan 1 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 6 - 7 + 2 = 1$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 1$ .

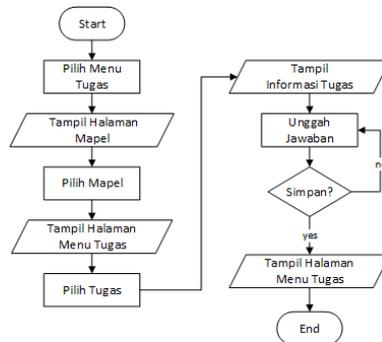
Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

**Tabel IV.14** *Test Case* Akses Materi oleh Siswa

<i>Path</i>	1
Jalur	1-2-3-4-5-6-7
Skenario	1. <i>Start</i> 2. Pilih menu materi 3. Tampil halaman mapel 4. Pilih mapel 5. Pilih judul materi 6. Tampil materi 7. <i>End</i>
Hasil Pengujian	Berhasil

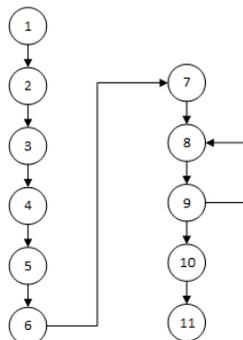
12. Pengujian *White Box* pada Akses Data Tugas oleh Siswa

a. *Flowchart*



**Gambar IV.50** *Flowchart* Akses Tugas oleh Siswa

b. *Flowgraph*



**Gambar IV.51** *Flowgraph* Akses Tugas oleh Siswa

Berdasarkan *flowgraph* akses tugas oleh siswa diatas terdapat 11 *edge*, 11 *node*, dan 2 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 11 - 11 + 2 = 2$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 2$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

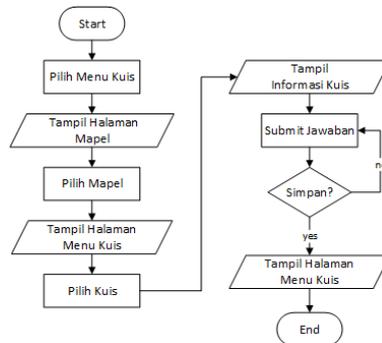
**Tabel IV.15** *Test Case* Akses Tugas oleh Siswa

<i>Path</i>	1
Jalur	1-2-3-4-5-6-7-8-9-10-11
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu tugas</li> <li>3. Tampil halaman mapel</li> <li>4. Pilih mapel</li> <li>5. Tampil halaman menu tugas</li> <li>6. Pilih tugas</li> <li>7. Tampil informasi tugas</li> <li>8. Unggah jawaban</li> <li>9. Simpan data sesuai</li> <li>10. Tampil halaman menu tugas</li> <li>11. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-5-6-7-8-9-8-9-10-11
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu tugas</li> </ol>

	<ol style="list-style-type: none"> <li>3. Tampil halaman mapel</li> <li>4. Pilih mapel</li> <li>5. Tampil halaman menu tugas</li> <li>6. Pilih tugas</li> <li>7. Tampil informasi tugas</li> <li>8. Unggah jawaban</li> <li>9. Simpan data tidak sesuai</li> <li>10. Unggah jawaban</li> <li>11. Simpan data sesuai</li> <li>12. Tampil halaman menu tugas</li> <li>13. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

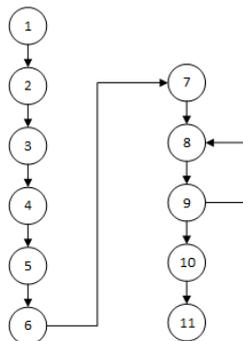
13. Pengujian *White Box* pada Akses Data Kuis oleh Siswa

a. *Flowchart*



**Gambar IV.52** *Flowchart* Akses Kuis oleh Siswa

b. *Flowgraph*



**Gambar IV.53** *Flowgraph* Akses Kuis oleh Siswa

Berdasarkan *flowgraph* akses kuis oleh siswa diatas terdapat 11 *edge*, 11 *node*, dan 2 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 11 - 11 + 2 = 2$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 2$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

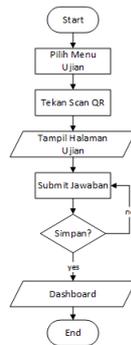
**Tabel IV.16** *Test Case* Akses Kuis oleh Siswa

<i>Path</i>	1
Jalur	1-2-3-4-5-6-7-8-9-10-11
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu kuis</li> <li>3. Tampil halaman mapel</li> <li>4. Pilih mapel</li> <li>5. Tampil halaman menu kuis</li> <li>6. Pilih kuis</li> <li>7. Tampil informasi kuis</li> <li>8. <i>Submit</i> jawaban</li> <li>9. Simpan data sesuai</li> <li>10. Tampil halaman menu kuis</li> <li>11. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-5-6-7-8-9-8-9-10-11
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu kuis</li> </ol>

	<ol style="list-style-type: none"> <li>3. Tampil halaman mapel</li> <li>4. Pilih mapel</li> <li>5. Tampil halaman menu kuis</li> <li>6. Pilih kuis</li> <li>7. Tampil informasi kuis</li> <li>8. <i>Submit</i> jawaban</li> <li>9. Simpan data tidak sesuai</li> <li>10. <i>Submit</i> jawaban</li> <li>11. Simpan data sesuai</li> <li>12. Tampil halaman menu kuis</li> <li>13. <i>End</i></li> </ol>
Hasil pengujian	Berhasil

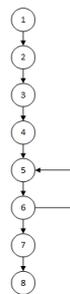
14. Pengujian *White Box* pada Akses Data Ujian oleh Siswa

a. *Flowchart*



**Gambar IV.54** *Flowchart* Akses Ujian oleh Siswa

b. *Flowgraph*



**Gambar IV.55** *Flowgraph* Akses Ujian oleh Siswa

Berdasarkan *flowgraph* akses ujian oleh siswa diatas terdapat 8 *edge*, 8 *node*, dan 2 *region* maka dapat dihitung nilai *cyclomatic complexity* menggunakan dengan cara, yaitu:

$$V(G) = E - N + 2$$

Dimana :

$E = \text{Edge}$

$N = \text{Node}$

Maka, persamaan dari rumus diatas adalah  $V(G) = 8 - 8 + 2 = 2$ , atau dapat menggunakan rumus  $V(G) = \text{region}$ , yang berarti  $V(G) = 2$ .

Dengan demikian hasil perhitungan *cyclomatic complexity* dapat dijelaskan dengan tabel berikut:

**Tabel IV.17** *Test Case* Akses Ujian oleh Siswa

<i>Path</i>	1
Jalur	1-2-3-4-5-6-7-8
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu ujian</li> <li>3. Tekan <i>scan QR</i></li> <li>4. Tampil halaman ujian</li> <li>5. <i>Submit</i> jawaban</li> <li>6. Simpan data sesuai</li> <li>7. <i>Dashboard</i></li> <li>8. <i>End</i></li> </ol>
Hasil Pengujian	Berhasil
<i>Path</i>	2
Jalur	1-2-3-4-5-6-5-6-7-8
Skenario	<ol style="list-style-type: none"> <li>1. <i>Start</i></li> <li>2. Pilih menu ujian</li> <li>3. Tekan <i>scan QR</i></li> <li>4. Tampil halaman ujian</li> <li>5. <i>Submit</i> jawaban</li> </ol>

	6. Simpan data tidak sesuai 7. <i>Submit</i> jawaban 8. Simpan data sesuai 9. <i>Dashboard</i> 10. <i>End</i>
Hasil pengujian	Berhasil

#### IV.2.3 Pengujian Efektivitas Penerapan Sistem *E-Learning*

Pengujian efektivitas penerapan sistem *e-learning* dilakukan dengan pengujian kuesioner yang terdiri dari 10 pertanyaan yang dibagikan kepada 20 responden, yang terdiri dari 17 siswa dan 3 guru SMP Negeri 2 Parepare. Kuesioner dibuat dengan menggunakan skala likert dari skala 1 sampai 5. Pengujian efektivitasnya dalam proses pembelajaran, efektif tidaknya dalam kecepatan dan ketepatan informasi sistemnya.

**Tabel IV.18** Nilai Skor Jawaban

Jawaban	Skor	Skor Maksimum
Sangat setuju	5	100
Setuju	4	80
Cukup Setuju	3	60
Kurang Setuju	2	40
Tidak Setuju	1	20

Setelah itu dapat dicari persentase masing-masing jawaban dengan menggunakan rumus:

$$Y = \frac{TS}{Skor\ Ideal} \times 100\%$$

Dimana,

Y = Nilai persentase keberhasilan

TS = Total skor responden =  $\sum$  skor \* responden

Skor ideal = skor \* jumlah responden = 5 \* 20 = 100

**Tabel IV.19** Kriteria Skor Berdasarkan Interval

Kategori	Keterangan
0% - 20%	Tidak Setuju
21%-40%	Kurang Setuju
41%-60%	Cukup Setuju
61%-80%	Setuju
81%-100%	Sangat Setuju

Kuesioner ini diujikan pada 20 responden, berikut hasil persentase setiap jawaban dan hitungan nilainya:

1. Pertanyaan 1

Sistem *e-learning* yang dibangun mudah untuk dipelajari.

**Tabel IV.20** Hasil Perhitungan Pertanyaan 1

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	12	60	$(92/100)*100$
Setuju	4	8	32	
Cukup Setuju	3	0	0	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	92	92%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 92% responden menilai sistem yang dibangun mudah untuk dipelajari.

2. Pertanyaan 2

Sistem *e-learning* yang dibangun mudah untuk digunakan.

**Tabel IV.21** Hasil Perhitungan Pertanyaan 2

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	9	45	$(86/100)*100$
Setuju	4	8	32	
Cukup Setuju	3	3	9	
Kurang Setuju	2	0	0	

Tidak Setuju	1	0	0	
Jumlah		20	86	86%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 86% responden menilai sistem yang dibangun mudah untuk digunakan.

### 3. Pertanyaan 3

Sistem *e-learning* yang dibangun memberikan manfaat bagi proses belajar mengajar di sekolah saya.

**Tabel IV.22** Hasil Perhitungan Pertanyaan 3

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	11	55	(88/100)*100
Setuju	4	6	24	
Cukup Setuju	3	3	9	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	88	88%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 88% responden menilai sistem yang dibangun memberikan manfaat bagi proses belajar mengajar di sekolah.

### 4. Pertanyaan 4

Sistem *e-learning* yang dibangun menghasilkan informasi yang akurat.

**Tabel IV.23** Hasil Perhitungan Pertanyaan 4

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	7	35	(87/100)*100
Setuju	4	13	52	
Cukup Setuju	3	0	0	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	87	87%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 87% responden menilai sistem yang dibangun menghasilkan informasi yang akurat.

5. Pertanyaan 5

Sistem *e-learning* yang dibangun menghasilkan informasi yang sesuai dengan kebutuhan belajar mengajar di sekolah saya.

**Tabel IV.24** Hasil Perhitungan Pertanyaan 5

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	14	70	(94/100)*100
Setuju	4	6	24	
Cukup Setuju	3	0	0	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	94	94%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 94% responden menilai sistem yang dibangun menghasilkan informasi yang sesuai dengan kebutuhan belajar mengajar di sekolah.

6. Pertanyaan 6

Sistem *e-learning* yang dibangun memberikan informasi yang *up-to-date*.

**Tabel IV.25** Hasil Perhitungan Pertanyaan 6

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	9	45	(89/100)*100
Setuju	4	11	44	
Cukup Setuju	3	0	0	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	89	89%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 89% responden menilai sistem yang dibangun memberikan informasi yang *up-to-date*.

7. Pertanyaan 7

Sistem *e-learning* yang dibangun bersifat *user-friendly*.

**Tabel IV.26** Hasil Perhitungan Pertanyaan 7

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	8	40	(86/100)*100
Setuju	4	10	40	
Cukup Setuju	3	2	6	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	86	86%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 86% responden menilai sistem yang dibangun bersifat *user-friendly*.

8. Pertanyaan 8

Sistem *e-learning* yang dibangun stabil dan lancar pengoperasiannya dalam proses belajar mengajar.

**Tabel IV.27** Hasil Perhitungan Pertanyaan 8

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	13	65	(92/100)*100
Setuju	4	6	24	
Cukup Setuju	3	1	3	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	92	92%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 92% responden menilai sistem yang dibangun stabil dan lancar pengoperasiannya dalam proses belajar mengajar.

9. Pertanyaan 9

Sistem *e-learning* yang dibangun mempermudah untuk mengevaluasi hasil belajar siswa.

**Tabel IV.28** Hasil Perhitungan Pertanyaan 9

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	6	30	(85/100)*100
Setuju	4	13	52	
Cukup Setuju	3	1	3	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	85	85%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 85% responden menilai sistem yang dibangun mempermudah untuk mengevaluasi hasil belajar siswa.

10. Pertanyaan 10

Sistem *e-learning* yang dibangun memberikan hasil tugas, kuis, dan ujian secara adil dan merata.

**Tabel IV.29** Hasil Perhitungan Pertanyaan 10

Jawaban	Skor	Responden	Jumlah Skor	Nilai Persentase
Sangat Setuju	5	10	50	(89/100)*100
Setuju	4	9	36	
Cukup Setuju	3	1	3	
Kurang Setuju	2	0	0	
Tidak Setuju	1	0	0	
Jumlah		20	89	89%

Berdasarkan nilai penyajian pernyataan dapat disimpulkan bahwa 86% responden menilai tugas, kuis, dan ujian yang diberikan melalui sistem yang dibangun memberikan nilai kepada siswa secara adil dan merata.

Rata-rata nilai persentase dari semua pertanyaan adalah sebagai berikut:

$$= \text{Total Nilai Persentase} / \text{Jumlah Pertanyaan}$$

$$= (92\% + 86\% + 88\% + 87\% + 94\% + 89\% + 86\% + 92\% + 85\% + 89\%) / 10$$

$$= 888\% / 10$$

$$= 88,8\%$$

Jadi, tingkat persetujuan terhadap aplikasi *e-learning* berdasarkan interval kriteria skor dimana rata-rata nilai persentase adalah 88,8% itu termasuk pada kategori “Sangat Setuju”.

#### IV.2.4 Implementasi API Website pada Aplikasi Android

Pada sub bab ini disajikan potongan *source code* program konfigurasi *controller* API pada *file* `apiController.php` yang berfungsi sebagai penghubung aplikasi *web* dengan aplikasi android.

##### 1. API Login

```
public function loginsiswa(Request $request){
    $email = $request->email;
    $password = $request->password;
    $cek = User::where('email', $email)->first();
    if($cek){
        if(Hash::check($password, $cek->password)){
            return response([
                'status' => 'OK',
                'message' => 'Berhasil Login',
                'id_siswa' => $cek->id,
                'nama_siswa' => $cek->name,
            ]);
        }else{
            return response([
                'status' => 'NO',
                'message' => 'Password Tidak Cocok'
            ]);
        }
    }else{
        return response([
            'status' => 'NO',
            'message' => 'Username Tidak di Temukan'
        ]);
    }
}
```

Potongan *source code* diatas berfungsi sebagai validasi login pada android seperti pada gambar IV.13, yang mana dari tampilan tersebut mengirim data berupa *email* dan *password* yang akan kemudian akan divalidasi dengan proses diatas . Keluarannya dari validasi tersebut akan diarahkan ke tampilan *dashboard* seperti pada gambar IV.15.

## 2. API Materi

```
public function materi($id){
    $datamapel = jadwal_kelas::where('users.id', $id)
    ->select('jadwal_kelas.*', 'mapels.id as idmapel',
    'mapels.nama_mapel as namamapel', 'siswas.id_user as
    idusersiswa', 'users.id as iduser', 'users.name as
    siswa', 'kelas.namakelas as namakelas', 'kelas.id as
    idkelas'
    )->leftJoin('mapels', function($join){
        $join->on('jadwal_kelas.mapel', '=', 'mapels.id');
    })->leftJoin('kelas', function($join){
        $join->on('jadwal_kelas.kelas', '=', 'kelas.id');
    })->leftJoin('siswas', function($join){
        $join->on('kelas.id', '=', 'siswas.kelas');
    })->leftJoin('users', function($join){
        $join->on('users.id', '=', 'siswas.id_user');
    })->get();

    return response()->json($datamapel);
}
```

Potongan *source code* diatas berfungsi untuk menampilkan tampilan *list* materi berdasarkan *id* siswa, dari data ini kemudian di konversi ke android berupa tampilan *list view*. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.18.

## 3. API Tampilan Materi

```
public function materimapel($id){
    $datamapel = Materi::where('mapels.id', $id)-
    >select('materis.id as idmateri', 'materis.judul as
    judul', 'materis.mapel as mapel', 'materis.urut as
    urut', 'materis.tingkat as tingkat',
    'materis.tahunajaran as tahunajaran',
    'materis.deskripsi as deskripsi', 'materis.file as
    file', 'mapels.id as id', 'mapels.guru as guru',
    'mapels.nama_mapel as nama_mapel'
    )->leftJoin('mapels', function($join){
        $join->on('materis.mapel', '=', 'mapels.id');
    })->get();

    return response()->json($datamapel);
}
```

Potongan *source code* diatas berfungsi untuk menampilkan tampilan detail materi dalam bentuk pdf berdasarkan *id* mapel, dari data ini kemudian di

konversi ke android berupa tampilan pdf materi. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.19.

#### 4. API Tugas

```
public function tugas($id){
    $datatugas = Tugas::where('users.id', $id)-
    >where('tugas.stat', 'B')->select('tugas.*',
    'siswas.*', 'users.name as namasiswa', 'mapels.*'
    )->leftJoin('siswas', function($join){
        $join->on('siswas.id', '=', 'tugas.siswa');
    })->leftJoin('users', function($join){
        $join->on('siswas.id_user', '=', 'users.id');
    })->leftJoin('kelas', function($join){
        $join->on('siswas.kelas', '=', 'kelas.id');
    })->leftJoin('jadwal_kelas', function($join){
        $join->on('jadwal_kelas.kelas', '=', 'kelas.id');
    })->leftJoin('mapels', function($join){
        $join->on('siswas.kelas', '=', 'kelas.id');
    })->get();
    return response()->json($datatugas);
}
```

Potongan *source code* diatas berfungsi untuk menampilkan *list* tugas berdasarkan *id* siswa, dari data ini kemudian di konversi ke android berupa tampilan *list view*. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.20.

#### 5. API Soal Tugas

```
public function tugaslist($siswa, $mapel){
    $datatugas = Tugas::where('users.id', $siswa)-
    >where('tugas.stat', 'B')->where('tugas.mapel',
    $mapel)->select('tugas.*', 'tugas.id as idTugas',
    'siswas.*', 'users.name as namasiswa', 'mapels.*'
    )->leftJoin('siswas', function($join){
        $join->on('siswas.id', '=', 'tugas.siswa');
    })->leftJoin('users', function($join){
        $join->on('siswas.id_user', '=', 'users.id');
    })->leftJoin('kelas', function($join){
        $join->on('siswas.kelas', '=', 'kelas.id');
    })->leftJoin('jadwal_kelas', function($join){
        $join->on('jadwal_kelas.kelas', '=', 'kelas.id');
    })->leftJoin('mapels', function($join){
        $join->on('siswas.kelas', '=', 'kelas.id');
    })->get();
}
```

```

        return response()->json($datatugas);
    }

```

Potongan *source code* diatas berfungsi untuk menampilkan detail atau soal dari tugas berdasarkan *id* siswa dan *id* mapel, dari data ini kemudian di konversi ke android berupa tampilan *list view*. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.21.

## 6. API Kuis

```

public function kuis($id){
    $datakuis = Kuis::where('users.id', $id)-
    >where('kuis.stat', 'B')->select('kuis.*', 'siswas.*',
    'users.name as namasiswa', 'mapels.*'
    )->leftJoin('siswas', function($join){
        $join->on('siswas.id', '=', 'kuis.siswa');
    })->leftJoin('users', function($join){
        $join->on('siswas.id_user', '=', 'users.id');
    })->leftJoin('kelas', function($join){
        $join->on('siswas.kelas', '=', 'kelas.id');
    })->leftJoin('jadwal_kelas', function($join){
        $join->on('jadwal_kelas.kelas', '=', 'kelas.id');
    })->leftJoin('mapels', function($join){
        $join->on('siswas.kelas', '=', 'kelas.id');
    })->get();

    return response()->json($datakuis);
}

```

Potongan *source code* diatas berfungsi untuk menampilkan *list* kuis berdasarkan *id* siswa, dari data ini kemudian di konversi ke android berupa tampilan *list view*. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.22.

## 7. API Soal Kuis

```

public function tampilsoal($id){
    $datatugas = Tugas::where('id', $id)->first();
    return view('soal', ['soal' => $datatugas-
    >deskripsi]);
}

```

Potongan *source code* diatas berfungsi untuk menampilkan detail atau soal dari kuis berdasarkan *id* siswa dan *id* mapel, dari data ini kemudian di

konversi ke android berupa tampilan *list view*. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.23.

#### 8. API Soal Ujian

```
public function soal($id){
    $dataujian = jadwal_ujian::select('soals.*')-
    >where('qr', $id)->
    leftJoin('soals', function($join){
        $join->on('soals.kelompoksoal', '=',
        'jadwal_ujians.id');
    })
    ->get();
    return response()->json($dataujian);
}
```

Potongan *source code* diatas berfungsi untuk menampilkan detail atau soal dari ujian berdasarkan *id* siswa, dari data ini kemudian di konversi ke android berupa tampilan *list view*. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.26.

#### 9. API QR Code

```
public function ujianqr($id){
    $datamapel = jadwal_ujian::where('qr', $id)
    ->select('jadwal_ujians.*', 'mapels.id as idmapel',
    'mapels.nama_mapel as namamapel'
    )->leftJoin('mapels', function($join){
        $join->on('jadwal_ujians.mapel', '=',
    'mapels.id');
    })->get();

    return response()->json($datamapel);
}
```

Potongan *source code* diatas berfungsi mengirim *code* qr kemudian diterjemahkan oleh API *service*, yang selanjutnya apabila *code* qr ditemukan maka akan tampil soal ujian dan jika tidak maka akan kembali ke *dashboard*. Hasil dari *source code* diatas dapat dilihat pada gambar IV.25.

#### 10. API Jadwal

```
public function jadwalkelas($id){
    $datamapel = jadwal_kelas::where('users.id', $id)
```

```

->select('jadwal_kelas.*', 'mapels.id as idmapel',
'mapels.nama_mapel as namamapel', 'siswas.id_user as
idusersiswa', 'users.id as iduser', 'users.name as
siswa', 'kelas.namakelas as namakelas', 'kelas.id as
idkelas'
)->leftJoin('mapels', function($join){
    $join->on('jadwal_kelas.mapel', '=', 'mapels.id');
})->leftJoin('kelas', function($join){
    $join->on('jadwal_kelas.kelas', '=', 'kelas.id');
})->leftJoin('siswas', function($join){
    $join->on('kelas.id', '=', 'siswas.kelas');
})->leftJoin('users', function($join){
    $join->on('users.id', '=', 'siswas.id_user');
})->get();

return response()->json($datamapel);
}

```

Potongan *source code* diatas berfungsi untuk menampilkan *list* jadwal berdasarkan *id* siswa, dari data ini kemudian di konversi ke android berupa tampilan tabel. Hasil API dari *source code* diatas dapat dilihat pada gambar IV.16.

## **BAB V**

### **PENUTUP**

#### **V.1 Kesimpulan**

Berdasarkan penulisan dan penelitian yang telah diuraikan, maka dapat ditarik beberapa kesimpulan sebagai berikut:

1. Terciptanya aplikasi *e-learning* yang dibuat dalam bentuk aplikasi *website* dan aplikasi android.
2. Dengan menggunakan API yang telah diciptakan dapat mempermudah akses data antar dua *platform* aplikasi *web* dan android, API ini juga meningkatkan fungsi dan penggunaan kedua aplikasi secara bersamaan.
3. Berdasarkan hasil pengujian kuesioner efektivitas penerapan sistem diperoleh nilai persentase sebesar 88,8% yang artinya penerapan aplikasi *e-learning* ini dari 20 responden menyatakan sangat setuju bahwa aplikasi ini efektif terhadap proses pembelajaran.

#### **V.2 Saran**

Berdasarkan hasil perancangan aplikasi ini, diberikan beberapa saran yang dapat membantu pengembangan aplikasi ini. Adapun beberapa saran yang diajukan yaitu:

1. Aplikasi *e-learning* dapat dikoneksikan dengan sistem informasi akademis agar memudahkan siswa atau guru dalam menggunakan sistem yang lebih terpadu.
2. Aplikasi *e-learning* dapat dikembangkan agar dapat berjalan diberbagai *platform*.

## DAFTAR PUSTAKA

- Achman, R. F., Witanti, W., & Hadiana, A. I. (2021). Pembangunan Perangkat Lunak E-Learning dalam Kegiatan Belajar Mengajar Sekolah Menengah Kejuruan Angkasa Husein. *Jurnal SAINTIKOM (Jurnal Sains Manajemen Informatika dan Komputer)*, 20(2), 130-138.
- Ade, A. P., & Novri, N. H. (2019). APLIKASI SIMPAN PINJAM PADA KOPERASI PT. TELKOM PALEMBANG (KOPEGTEL) MENGGUNAKAN Andrian, D.(2021). Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Pengawasan Proyek Berbasis Web. *Jurnal Informatika Dan Rekayasa Perangkat Lunak (JATIKA)*, 2(1).
- Adibhadiansyah, M. (2016). Pengembangan Sistem Informasi Kos Berbasis Android. *Jurnal Manajemen Informatika*, 5(2).
- Agustini, A., & Kurniawan, W. J. (2020). Sistem E-Learning Do'a dan Iqro' dalam Peningkatan Proses Pembelajaran pada TK Amal Ikhlas. *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi (JMApTeKsi)*, 1(3), 154-159.
- Arianto, M. A. (2016). Analisis dan Perancangan Representational State Transfer (REST) Web Service Sistem Informasi Akademik STT Terpadu Nurul Fikri Menggunakan Yii Framework. *Jurnal Teknologi Terpadu*, 2(2).
- Azhar, D. (2019). *Aplikasi E-Learning Siswa Berbasis Web Pada SMPN Bernas Kabupaten Pelalawan Riau (Studi Kasus: SMPN Bernas Kab. Pelalawan Riau)* (Doctoral dissertation, University of Technology Yogyakarta).
- Dharwiyanti, S., & Wahono, R. S. (2003). Pengantar Unified Modeling Language (UML). *IlmuKomputer.com*, 1-13.
- Galing, H. A., & Lestari, A. D. (2010). *Pembuatan Aplikasi Widget Untuk Monitoring Saham* (Doctoral dissertation, University of AMIKOM Yogyakarta).
- Gunawan, V. A., Karliani, E., Triyani, T., Saefulloh, A., & Putra, L. S. A. (2021). Desain Fitur Aplikasi E-Learning Penunjang Pembelajaran Berbasis Android. *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, 7(3), 314-321.
- Hadisi, L., & Muna, W. (2015). Pengelolaan teknologi informasi dalam menciptakan model inovasi pembelajaran (e-learning). *Al-TA'DIB: Jurnal Kajian Ilmu Kependidikan*, 8(1), 117-140.

- Hoetamy, D. M., Nugroho, A., & Harris, A. (2020). Perancangan E-Learning Pada SMK Veteran Kota Jambi Berbasis Android. *Jurnal Ilmiah Mahasiswa Teknik Informatika*, 2(2), 172-185.
- Jimi, A. (2020). Perancangan Sistem E-Learning Berbasis Web Pada Smp N 2 Busalangga. *Jurnal Pendidikan Teknologi Informasi (JUKANTI)*, 3(1), 29-37.
- Juansyah, A. (2015). Pembangunan aplikasi child tracker berbasis assisted–global positioning system (a-gps) dengan platform android. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 1(1), 1-8.
- Khamidah, K., & Triyono, R. A. (2013). Pengembangan aplikasi e-learning berbasis web dengan php dan my sql studi kasus smpn 1 arjosari. *IJNS-Indonesian Journal on Networking and Security*, 2(2).
- Krisnoanto, A. (2018). *Penerapan Metode User Centered Design Pada Aplikasi E-Learning Berbasis Android (Studi Kasus: SMAN 3 Sidoarjo)* (Doctoral dissertation, Universitas Brawijaya).
- Kurniawan, H., Apriliah, W., Kurniawan, I., & Firmansyah, D. (2020). Penerapan Metode *Waterfall* Dalam Perancangan Sistem Informasi Penggajian Pada Smk Bina Karya Karawang. *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, 14(4), 159-169.
- Kuswanto, J., & Radiansah, F. (2018). Media Pembelajaran Berbasis Android Pada Mata Pelajaran Sistem Operasi Jaringan Kelas XI. *Jurnal Media Infotama*, 14(1).
- Mediana, D. (2018). Rancang Bangun Aplikasi Helpdesk (A-Desk) Berbasis Web Menggunakan Framework Laravel (Studi Kasus Di Pdam Surya Sembada Kota Surabaya) Design Build Helpdesk Applications (A-Desk) Based Web Using Framework Laravel (Case Study in Surya Sembada Municipal Waterwork Of Surabaya). *Jurnal Manajemen Informatika*, 8(2).
- Mubarak, A. (2019). Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek. *JIKO (Jurnal Informatika dan Komputer)*, 2(1), 19-25.
- Mutia, I., & LEONARD, L. (2015). Kajian penerapan e-learning dalam proses pembelajaran di perguruan tinggi. *Faktor Exacta*, 6(4), 278-289.
- Nuryazid. (2017). Perancangan Sistem E-Learning Berbasis Web Untuk Peningkatan Kualitas Pembelajaran Pada SMK Mathla'Ulanwar. Skripsi. Tidak Diterbitkan. Jurusan Teknik Informatika. STIMIK RAHARJA: Tangerang.

- Pahlevi, O., Mulyani, A., & Khoir, M. (2018). Sistem Informasi Inventori Barang Menggunakan Metode Object Oriented Di Pt. Livaza Teknologi Indonesia Jakarta. *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, 5(1).
- Pasaribu, J. S. (2017). Penerapan framework yii pada pembangunan sistem ppdb smp bppi baleendah kabupaten bandung. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 3(2).
- Salehudin, H., Fatimah, F., & Riana, F. (2019, October). Rancang Bangun Sistem Pembelajaran Berbasis E-Learning di Sekolah Menengah Pertama (SMP) Triple “J”. In *Seminar Nasional Teknologi Informasi* (Vol. 2, pp. 238-243).
- Saragih, R. R. (2016). Pemrograman dan bahasa Pemrograman. *Praktek Otomasi Perkantoran*.
- Sari, T. T., & Cahyono, A. H. (2020). Pengembangan E-Learning Berbasis Android “Fun Math” Sebagai Alternatif Belajar Matematika di Tengah Pandemi. *Jurnal Cendekia: Jurnal Pendidikan Matematika*, 4(2), 1283-1298.
- Sibarani, S. A. P., Napitupulu, J., & Jamaluddin, J. (2017). Aplikasi Pengolahan Data Alumni Diploma Tiga Manajemen Informatika Universitas Methodist Indonesia Medan. *METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi*, 1(1), 39-45.
- Suartama, I. K. (2014). E-Learning Konsep dan Aplikasinya. Singaraja: Undiksha Press. Tersedia dari [https://www.researchgate.net/publication/337136881\\_E-Learning\\_Konsep\\_dan\\_Aplikasinya](https://www.researchgate.net/publication/337136881_E-Learning_Konsep_dan_Aplikasinya)
- Susanto, H., Widiartin, T., & Pratama, F. H. S. (2016). Aplikasi Pembelajaran Berbasis Android (E-Learning) Di MA. Daruttaqwa Gresik. *Melek IT Information Technology Journal*, 2(2), 81-88.
- Susanto, W. E., & Astuti, Y. G. A. (2017). Perancangan E-Learning Berbasis Web Pada SMP Negeri 3 Patuk Gunungkidul Yogyakarta. *Bianglala Informatika*, 5(2), 62-69.
- Suteja, B. R., & Harjoko, A. (2008). User *Interface* Design for e-Learning System. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)* (Vol. 1, No. 1).
- Tjandra, S., & Chandra, G. S. (2020). Pemanfaatan Flutter dan Electron Framework pada Aplikasi Inventori dan Pengaturan Pengiriman Barang. *Journal of Information System, Graphics, Hospitality and Technology*, 2(02), 76-81.

- Wati, E. F., & Kusumo, A. A. (2016). Penerapan Metode Unified Modeling Language (UML) Berbasis Dekstop pada Sistem Pengolahan Kas Kecil Studi Kasus pada PT Indo Mada Yasa Tangerang. *Syntax: Jurnal Informatika*, 5(1), 24-36.
- Wibawa, J. C., & Edah, E. (2015). Aplikasi E-Learning di SMP Negeri 46 Bandung. *Jurnal Teknologi dan Informasi*, 5(1), 1-10.
- Yunianto, A. (2021). *SISTEM INFORMASI PENJUALAN MATERIAL ALAM BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL (STUDI KASUS: UD. MANUNGGAL)* (Doctoral dissertation, STMIK AKAKOM).

# LAMPIRAN

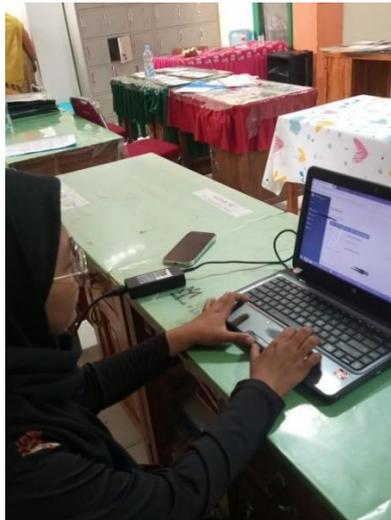
## 1. Wawancara dan Observasi

Wawancara dan observasi dengan wakasek kesiswaan.



## 2. Admin

Pengujian aplikasi dan pengisian kuesioner pada admin



### 3. Guru

Pengujian aplikasi dan pengisian kuesioner pada guru



### 4. Siswa

Pengujian aplikasi dan pengisian kuesioner pada siswa/i



## 5. Pertanyaan Kuesioner via Google Form

Kuesioner dapat diakses dengan link <https://forms.gle/Qc4jJhAHmYJFN7Hr9>

**KUESIONER PENELITIAN "APLIKASI E-LEARNING BERBASIS ANDROID PADA SMP NEGERI 2 PAREPARE"**

Dalam rangka pemenuhan tugas akhir, Saya, Fandy Febriansyah bermaksud melakukan penelitian ilmiah untuk penyusunan tugas akhir dengan judul "Aplikasi E-Learning Berbasis Android pada SMP Negeri 2 Parepare". Sehubungan dengan hal tersebut Saya sangat mengharapkan kesediaan Bapak/Ibu/Saudara/Saudari untuk meluangkan waktunya seponok untuk mengisi beberapa pertanyaan pada kuesioner ini. Atas perhatiannya dan kerja samanya, saya ucapkan terima kasih.

fandyfebriansyah2@gmail.com (not shared) Switch account

\* Required

**Nama \***  
Your answer

**NIS/NIP**  
Your answer

**Kelas**  
Your answer

**Jenis Kelamin \***  
 Laki - Laki  
 Perempuan

**Usia \***  
Your answer

1. Sistem e-learning yang dibangun mudah untuk dipelajari \*  
Choose

2. Sistem e-learning yang dibangun mudah untuk digunakan \*  
Choose

3. Sistem e-learning yang dibangun memberikan manfaat bagi proses belajar mengajar di sekolah saya \*  
Choose

4. Sistem e-learning yang dibangun menghasilkan informasi yang akurat \*  
Choose

5. Sistem e-learning yang dibangun menghasilkan informasi yang sesuai dengan \* kebutuhan belajar mengajar di sekolah saya  
Choose

6. Sistem e-learning yang dibangun memberikan informasi yang up-to-date \*  
Choose

7. Sistem e-learning yang dibangun bersifat user-friendly \*  
Choose

8. Sistem e-learning yang dibangun stabil dan lancar pengoperasiannya dalam \* proses belajar mengajar  
Choose

9. Sistem e-learning yang dibangun mempermudah untuk mengevaluasi hasil \* belajar siswa  
Choose

10. Sistem e-learning yang dibangun memberikan hasil tugas, kuis, dan ujian \* secara adil dan merata  
Choose

Submit Clear form

6. Jawaban per Pertanyaan Kuesioner via Google Form

a. Pertanyaan 1

Dari 20 responden diperoleh 12 yang sangat setuju dan 8 yang setuju.



b. Pertanyaan 2

Dari 20 responden diperoleh 9 yang sangat setuju, 8 setuju, dan 3 cukup setuju.



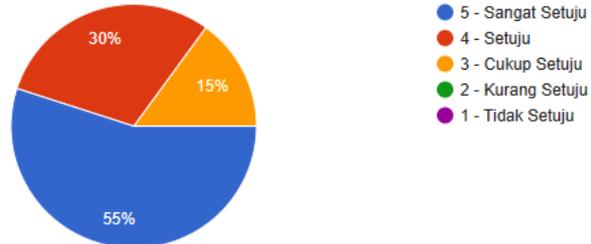
c. Pertanyaan 3

Dari 20 responden diperoleh 11 yang sangat setuju, 6 setuju, 3 cukup setuju.

3. Sistem e-learning yang dibangun memberikan manfaat bagi proses belajar mengajar di sekolah saya

 Copy

20 responses



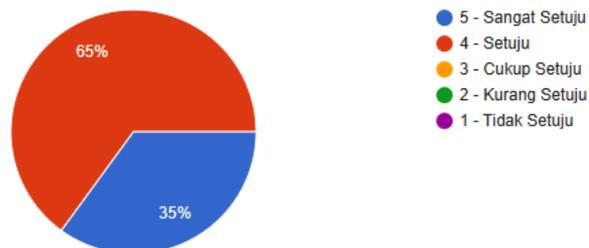
d. Pertanyaan 4

Dari 20 responden diperoleh 7 yang sangat setuju dan 13 yang setuju.

4. Sistem e-learning yang dibangun menghasilkan informasi yang akurat

 Copy

20 responses



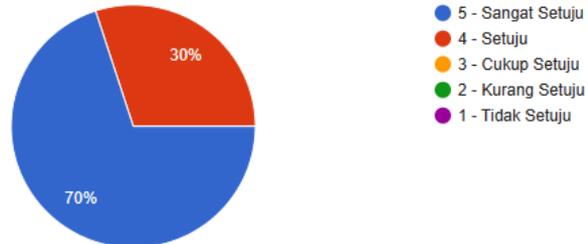
e. Pertanyaan 5

Dari 20 responden diperoleh 14 yang sangat setuju dan 6 yang setuju.

5. Sistem e-learning yang dibangun menghasilkan informasi yang sesuai dengan kebutuhan belajar mengajar di sekolah saya

 Copy

20 responses



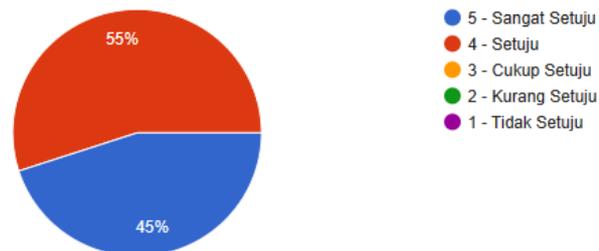
f. Pertanyaan 6

Dari 20 responden diperoleh 9 yang sangat setuju dan 11 yang setuju.

6. Sistem e-learning yang dibangun memberikan informasi yang up-to-date

 Copy

20 responses



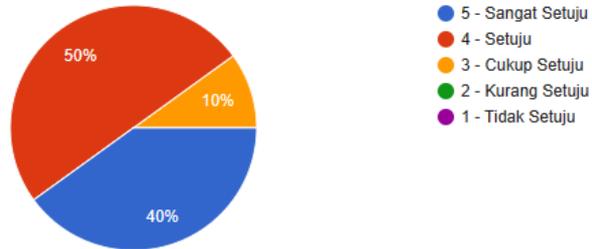
g. Pertanyaan 7

Dari 20 responden diperoleh 8 yang sangat setuju, 10 setuju, 2 cukup setuju.

7. Sistem e-learning yang dibangun bersifat user-friendly

 Copy

20 responses



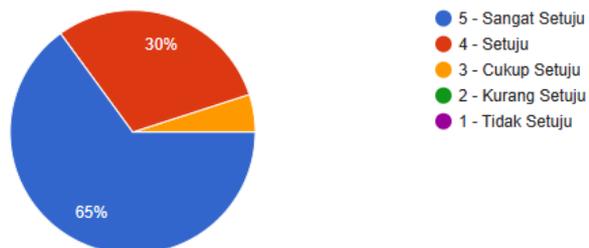
h. Pertanyaan 8

Dari 20 responden diperoleh 13 yang sangat setuju, 6 setuju, 1 cukup setuju.

8. Sistem e-learning yang dibangun stabil dan lancar pengoperasiannya dalam proses belajar mengajar

 Copy

20 responses

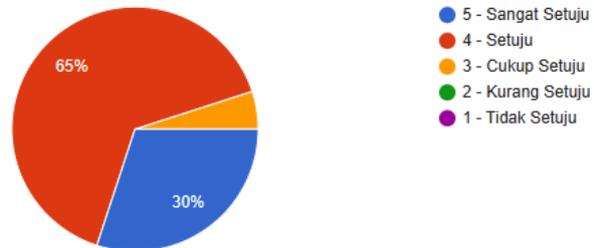


i. Pertanyaan 9

Dari 20 responden diperoleh 6 yang sangat setuju, 13 setuju, 1 cukup setuju.

9. Sistem e-learning yang dibangun mempermudah untuk mengevaluasi hasil belajar siswa [Copy](#)

20 responses



j. Pertanyaan 10

Dari 20 responden diperoleh 10 yang sangat setuju, 9 setuju, 1 cukup setuju.

10. Sistem e-learning yang dibangun memberikan hasil tugas, kuis, dan ujian secara adil dan merata [Copy](#)

20 responses

