

**PENGEMBANGAN MODUL INFORMASI PUBLIK DAN MODUL EVALUASI
PADA SISTEM INFORMASI AKADEMIK SMPN 29 BANDAR LAMPUNG
BERBASIS *WEBSITE* MENGGUNAKAN *FRAMEWORK* CODEIGNITER 4
DENGAN METODE *EXTREME PROGRAMMING***

(Skripsi)

Oleh

**ENJELITA AINI NATASYA
NPM 2117051028**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

**PENGEMBANGAN MODUL INFORMASI PUBLIK DAN MODUL EVALUASI
PADA SISTEM INFORMASI AKADEMIK SMPN 29 BANDAR LAMPUNG
BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK CODEIGNITER 4
DENGAN METODE *EXTREME PROGRAMMING***

Oleh

ENJELITA AINI NATASYA

Skripsi

**Sebagai Salah Satu Syarat Untuk Mencapai Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

ABSTRAK

PENGEMBANGAN MODUL INFORMASI PUBLIK DAN MODUL EVALUASI PADA SISTEM INFORMASI AKADEMIK SMPN 29 BANDAR LAMPUNG BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK CODEIGNITER 4 DENGAN METODE EXTREME PROGRAMMING

Oleh

ENJELITA AINI NATASYA

Penelitian ini bertujuan mengembangkan sistem informasi akademik berbasis website di SMPN 29 Bandar Lampung menggunakan framework CodeIgniter 4. Sistem ini mencakup modul informasi publik dan evaluasi untuk mempermudah akses informasi dan pemantauan hasil belajar secara real-time. Modul informasi publik menyajikan berbagai data sekolah secara terpusat dan transparan, sedangkan modul evaluasi dilengkapi fitur e-raport dan ranking. Metode pengembangan menggunakan Extreme Programming (XP) karena fleksibilitasnya terhadap perubahan. Pengujian dilakukan dengan metode Black-Box menggunakan teknik Equivalence Partitioning untuk fungsionalitas, serta User Acceptance Testing (UAT) untuk aspek non-fungsional. Hasil menunjukkan seluruh fungsi berjalan sesuai dengan spesifikasi, dan tingkat keberhasilan UAT mencapai 86%. Sistem ini diharapkan dapat meningkatkan efisiensi, akurasi, dan transparansi dalam pengelolaan informasi akademik. Penerapan sistem ini juga menjadi langkah awal transformasi digital di lingkungan sekolah. Dengan adanya sistem ini, penyampaian informasi menjadi lebih cepat, mudah, dan terstruktur.

Kata kunci: Sistem Informasi Akademik, CodeIgniter 4, Extreme Programming.

ABSTRACT

DEVELOPMENT OF PUBLIC INFORMATION AND EVALUATION MODULES IN THE WEB-BASED ACADEMIC INFORMATION SYSTEM OF SMPN 29 BANDAR LAMPUNG USING THE CODEIGNITER 4 FRAMEWORK AND THE EXTREME PROGRAMMING METHOD

By

ENJELITA AINI NATASYA

This study aims to develop a web-based academic information system at SMPN 29 Bandar Lampung using the CodeIgniter 4 framework. The system includes a public information module and an evaluation module to facilitate information access and real-time monitoring of student learning outcomes. The public information module provides centralized and transparent school data, while the evaluation module features e-report cards and ranking. The development method used is Extreme Programming (XP) due to its flexibility in adapting to changes. Testing was conducted using the Black-Box method with the Equivalence Partitioning technique for functional aspects, and User Acceptance Testing (UAT) for non-functional aspects. The results show that all functions operate according to specifications, with a UAT success rate of 86%. This system is expected to improve efficiency, accuracy, and transparency in academic information management. The implementation also represents an initial step toward digital transformation within the school environment. With this system, information delivery becomes faster, easier, and more structured.

Keywords: Academic Information System, CodeIgniter 4, Extreme Programming.

Judul Skripsi

: **PENGEMBANGAN MODUL
INFORMASI PUBLIK DAN MODUL
EVALUASI PADA SISTEM INFORMASI
AKADEMIK SMPN 29 BANDAR
LAMPUNG BERBASIS WEBSITE
MENGUNAKAN FRAMEWORK
CODEIGNITER 4 DENGAN METODE
EXTREME PROGRAMMING**

Nama Mahasiswa

: **Enjelita Aini Natasya**

Nomor Pokok Mahasiswa

: 2117051028

Program Studi

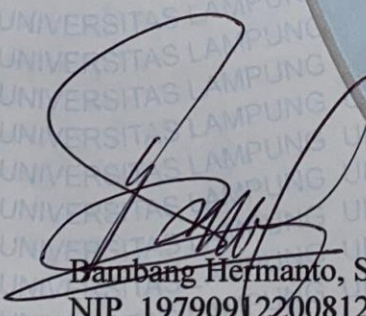
: Ilmu Komputer

Fakultas

: Matematika dan Ilmu Pengetahuan Alam

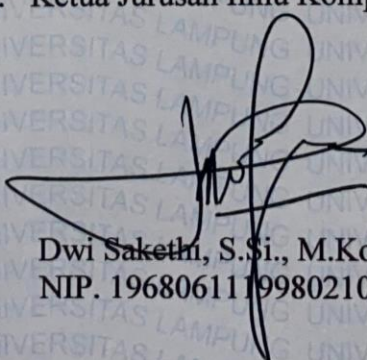


1. Komisi Pembimbing


Bambang Hermanto, S.Kom., M.Cs
NIP. 197909122008121002


Muhaqiqin, S.Kom., M.T.I.
NIP. 199305252022031009

2. Ketua Jurusan Ilmu Komputer


Dwi Sakethi, S.Si., M.Kom
NIP. 196806111998021001

MENGESAHKAN

1. Tim Penguji

Ketua : Bambang Hermanto, S.Kom., M.Cs

Sekretaris : Muhaqiqin, S.Kom., M.T.I.

Penguji Utama : Ossy Dwi Endah Wulansari, S.SI, M.Si

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Ing. Heri Satria S.Si., M.Si.

NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: 20 Juni 2025

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Enjelita Aini Natasya

NPM : 2117051028

Menyatakan bahwa skripsi saya yang berjudul **"Pengembangan Modul Informasi Publik dan Modul Evaluasi Pada Sistem Informasi Akademik SMPN 29 Bandar Lampung Berbasis Website Menggunakan Framework Codeigniter 4 Dengan Metode Extreme Programming"** merupakan karya saya sendiri dan bukan merupakan karya orang lain. Semua isi tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 30 Juni 2025



Enjelita Aini Natasya
NPM. 2117051028

RIWAYAT HIDUP



Penulis dilahirkan di Bandar Lampung pada tanggal 18 Agustus 2003 sebagai anak pertama dari 2 bersaudara, dari Bapak Nurul Huda, S.E. dan Ibu Oktavianti, S.H. Penulis telah menyelesaikan pendidikan formal di SDN 2 Wayhalim pada tahun 2015. SMPN 1 Bandar Lampung pada tahun 2018. Dan SMA YP UNILA pada tahun 2021. Penulis terdaftar di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas lampung melalui jalur masuk SNMPTN pada tahun 2021.

Selama menjadi mahasiswa, penulis aktif mengikuti beberapa kegiatan antara lain:

1. Menjadi Anggota bidang Keilmuan Himpunan Mahasiswa Ilmu Komputer Universitas Lampung pada tahun 2022.
2. Menjadi Koordinator Divisi Konsumsi pada kegiatan Workshop Wawancara Kerja pada tahun 2022.
3. Menjadi Asisten Dosen mata kuliah Pemrograman Web Dasar Pada tahun 2024.
4. Mengikuti kegiatan kerja praktik di Bidang Perbendaharaan Badan Pengelola Keuangan dan Aset Daerah (BPKAD) pada tahun 2023.
5. Mengikuti kegiatan Kuliah Kerja Nyata (KKN) Universitas Lampung Periode 2 di Desa Karya Tani pada tahun 2024.

MOTTO

*" Karena sesungguhnya sesudah kesulitan itu ada kemudahan, sesungguhnya
sesudah kesulitan itu ada kemudahan."*

(Q.S Al Insyirah: 5-6)

*" Barangsiapa yang mengerjakan kebaikan sekecil apapun, niscaya dia akan
melihat (balasan)nya. "*

(Q.S Al-Zalzalah: 7)

*"And it's fine to fake it 'til you make it
'Til you do, 'til it's true"*

"They laughed. I worked. And now, they watch."

PERSEMBAHAN

Alhamdulillahirobbilalamin

Puji syukur kepada Allah SWT yang selalu memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan penulisan skripsi ini. Shalawat serta salam selalu disanjungkan kepada Nabi Muhammad SAW

Kupersembahkan karya ini kepada:

Mama Tercinta

Yang selalu memberikan dukungan. terimakasih atas segala pengorbanan yang dilakukan, do'a yang selalu dipanjatkan, dan perhatian yang selalu diberikan.

Seluruh Keluarga Besar Ilmu Komputer 2021

Almamater Tercinta, Universitas Lampung

SANWACANA

Puji syukur atas kehadiran Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pengembangan Modul Informasi Publik dan Modul Evaluasi Pada Sistem Informasi Akademik SMPN 29 Bandar Lampung Berbasis Website Menggunakan Framework Codeigniter 4 Dengan Metode Extreme Programming” telah diselesaikan. Tidak lupa shalawat dan salam yang senantiasa dicurahkan kepada baginda Nabi Muhammad SAW yang menjadi suri teladan kepada seluruh umatnya. Pada kesempatan ini penulis mengucapkan terima kasih kepada semua pihak yang telah membantu dalam penyusunan skripsi ini, yaitu:

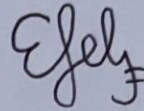
1. Allah SWT yang telah memberikan berkah, hidayah, rahmat, serta karunia-Nya selama proses penyelesaian skripsi ini.
2. Ibu Oktavianti, S.H. selaku mama tercinta dan Muhammad Raju Jan Septa selaku adik tersayang, yang senantiasa yang selalu memberikan semangat, dukungan, dan do'a serta membantu dalam segala hal yang tidak terhitung nilainya.
3. Bapak Nurul Huda, S.E. selaku bapak yang senantiasa memberi semangat, dukungan, dan do'a serta membantu dalam segala hal.
4. Bapak Dr. Eng. Heri Satria S.Si., M.Si. selaku Dekan Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
5. Bapak Dwi Sakethi, S.Si., M.Kom. selaku Ketua Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
6. Ibu Yunda Heningtyas, M.Kom. selaku Sekretaris Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
7. Bapak Febi Eka Febriansyah, M.T selaku Dosen Pembimbing Akademik yang telah memberikan saran dan dukungan kepada penulis.
8. Bambang Hermanto, S.Kom., M.Cs. selaku Dosen Pembimbing

utama dan Bapak Muhaqiqin, S.Kom., M.T.I. atas ketersediannya dalam memberikan bimbingan, motivasi, kritik, serta saran dalam proses penyelesaian skripsi.

9. Ibu Ossy Dwi Endah Wulansari, S.SI, M.Si. selaku Dosen Pembahas yang telah memberikan kritik dan saran hingga masukan kepada penulis selama proses penyelesaian skripsi.
10. Bapak dan Ibu Dosen Jurusan Ilmu Komputer Universitas Lampung yang telah memberikan ilmu, pengetahuan serta pengalamannya kepada penulis selama penulis menjadi mahasiswa.
11. Seluruh staf Jurusan Ilmu Komputer yang selalu membantu penulis dari awal hingga akhir masa perkuliahan.
12. Muhammad Rafi Satria yang telah kebersamai penyelesaian skripsi dari awal hingga akhir masa perkuliahan.
13. Teman-Teman "Tebu Tubaba/ Asiknyaaa" yaitu, Annisa, Tasya, Nathan, Muca, Peem, Zahra, Ipeh, Wirda, Rafi, Iqbal, Hanip, Zidan, Reja, Alfa, Qolby, Bagas, Rimuru selaku rekan seperjuangan di awal masa perkuliahan.
14. Teman-Teman "Sahabat Jannah" yaitu, Adhis, Tasya, Talia, Nindi, April, Ika, Siska, Yuna, Ayuni selaku rekan seperjuangan di akhir masa perkuliahan.
15. Teman-Teman SMP yang sudah bersama selama 10 tahun ini, Zulfa, Bella, Annisa, Puja, Steven.
16. Teman-Teman yang menjadi keluarga besar jurusan Ilmu Komputer'21 selama menjalankan masa studi di Jurusan Ilmu Komputer Universitas Lampung.

Proses dalam penyusunan skripsi ini tentunya terdapat banyak kekurangan dikarenakan keterbatasan pengetahuan serta pengalaman. Semoga skripsi ini dapat membawa manfaat dan keberkahan bagi perkembangan ilmu pengetahuan terutama bagi seluruh civitas jurusan Ilmu Komputer Universitas Lampung.

Bandar Lampung, 30 Juni 2025



Enjelita Aini Natasya
NPM. 2117051028

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

I. PENDAHULUAN

1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	4

II. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu	5
2.1.1 Pengembangan Sistem Informasi Akademik Berbasis Web di SMP Negeri 1 Sukoharjo	5
2.1.2 Penerapan Metode Programing Dalam Sistem Informasi Akademik SDN Naula Teladan	6
2.1.3 Sistem Informasi Akademik Berbasis Web (Studi Kasus: SMA AVICENNA)	7
2.1.4 Model Pengembangan Sistem Informasi Akademik Berbasis Web Cendekia: Desain Menemukan Prinsip-prinsip Pustaka	7
2.1.5 Pengembangan Sistem Informasi Akademik Berbasis Mobile Menggunakan Pustaka di Universitas Negeri Semarang	7
2.1.6 Sistem Informasi Akademik Berbasis Web di Pustaka Pustaka	8
2.2 Teori dan Konsep Dasar	9
2.2.1 Pengertian Sistem Informasi Akademik	9

DAFTAR ISI

	Halaman
DAFTAR ISI	i
DAFTAR GAMBAR	iii
DAFTAR TABEL	vi
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
II. TINJAUAN PUSAKA	5
2.1 Penelitian Terdahulu	5
2.1.1 Pengembangan Sistem Informasi Akademik berbasis <i>Web</i> di SMP Negeri 3 Sidikalang.....	5
2.1.2 Penerapan <i>Extreme Programming</i> Dalam Sistem Informasi Akademik SDN Kuala Teladas	6
2.1.3 Sistem Informasi Akademik Berbasis <i>Website</i> (Studi Kasus: SMPIT AVICENNA).....	6
2.1.4 Model Pengembangan Sistem Informasi Akademik Berbasis User Centered Design Menerapkan Framework Flask Python.....	7
2.1.5 Pengembangan Sistem Informasi Akademik Berbasis Mobile Menggunakan Flutter di Universitas Narotama Surabaya	7
2.1.6 Sistem Informasi Akademik Santri Berbasis Web di Pondok Pesantren	8
2.2 Uraian Landasan Teori.....	9
2.2.1 Pengertian Sistem Informasi Akademik	9

2.2.2	Pengertian PHP	9
2.2.3	Pengertian <i>Extreme Programming</i> (XP).....	10
2.2.4	Pengertian <i>Unified Modeling Language</i> (UML).....	11
2.2.5.	<i>Class Diagram</i>	14
2.2.6.	Pengertian <i>CodeIgniter</i> 4	15
2.2.7.	Pengertian <i>User Acceptance Testing</i> (UAT).....	15
2.2.8.	Pengertian <i>Black Box Testing</i>	16
III.	METODOLOGI PENELITIAN	17
3.1.	Waktu dan Tempat Pelaksanaan	17
3.2.	Tahapan Penelitian	17
3.2.1.	Tahap Pengumpulan Data	18
3.2.2.	Tahap Pengembangan Sistem	20
3.2.3.	Tahap Penulisan Laporan.....	90
IV.	HASIL DAN PEMBAHASAN	91
4.1.	Hasil	91
4.2.	Pembahasan.....	92
4.2.1.	Iterasi.....	92
4.2.2.	Tahap Pengujian.....	135
V.	KESIMPULAN DAN SARAN	155
5.1.	Kesimpulan	155
5.2.	Saran.....	155
	DAFTAR PUSTAKA	157

DAFTAR GAMBAR

Gambar	Halaman
1. Alur Tahapan Metode Extreme Programming	10
2. Tahapan Penelitian	18
3. <i>Use Case Diagram</i>	25
4. CRC Cards	37
5. Halaman Beranda	39
6. Halaman Sejarah Singkat	40
7. Halaman Visi Misi	40
8. Halaman Struktur Organisasi	41
9. Halaman Hymne.....	41
10. Halaman Galeri Kegiatan	42
11. Halaman Prestasi	42
12. Halaman Ekstrakurikuler	43
13. Halaman <i>Ranking</i>	44
14. Halaman <i>E-Raport</i>	44
15. <i>Class Diagram</i>	45
16. <i>Activity Diagram</i> Melihat Visi Misi.....	48
17. <i>Activity Diagram</i> Melihat Struktur Organisasi	49
18. <i>Activity Diagram</i> Melihat Hymne.....	50
19. <i>Activity Diagram</i> Melihat Sejarah Singkat	51
20. <i>Activity Diagram</i> Melihat Prestasi	52
21. <i>Activity Diagram</i> Melihat Galeri Kegiatan	53
22. <i>Activity Diagram</i> Melihat Ekstrakurikuler.....	54
23. <i>Activity Diagram</i> Melihat <i>Ranking</i>	55
24. <i>Activity Diagram</i> Melihat E-Raport.....	56

25. <i>Activity Diagram</i> Menambah Prestasi.....	58
26. <i>Activity Diagram</i> Mengubah Prestasi	59
27. <i>Activity Diagram</i> Menghapus Prestasi	61
28. <i>Activity Diagram</i> Menambah Ekstrakurikuler	62
29. <i>Activity Diagram</i> Mengubah Ekstrakurikuler	63
30. <i>Activity Diagram</i> Menghapus Ekstrakurikuler.....	64
31. <i>Activity Diagram</i> Menambah Sejarah Singkat.....	66
32. <i>Activity Diagram</i> Mengubah Sejarah Singkat.....	67
33. <i>Activity Diagram</i> Menghapus Sejarah Singkat	68
34. <i>Activity Diagram</i> Menambah Galeri Kegiatan.....	70
35. <i>Activity Diagram</i> Mengubah Galeri Kegiatan.....	72
36. <i>Activity Diagram</i> Menghapus Galeri Kegiatan	73
37. Halaman <i>Ranking</i>	93
38. Halaman Detail E-Raport	95
39. Halaman Detail E-Raport (Lanjutan)	96
40. Halaman Visi Misi	98
41. Halaman Struktur Organisasi	100
42. Halaman Hymne.....	101
43. Halaman Sejarah Singkat	103
44. Halaman Tambah Data Sejarah Kepemimpinan	105
45. Halaman Ubah Data Sejarah Kepemimpinan	106
46. Halaman Hapus Sejarah Kepemimpinan	109
47. Halaman Prestasi.....	111
48. Halaman Tambah Prestasi.....	112
49. Halaman Ubah Prestasi	114
50. Halaman Hapus Prestasi.....	117
51. Halaman <i>Ekstrakurikuler</i>	119
52. Halaman <i>Tambah Ekstrakurikuler</i>	121
53. Halaman <i>Ubah Ekstrakurikuler</i>	122
54. Halaman <i>Hapus Ekstrakurikuler</i>	125
55. Halaman Galeri Kegiatan	127

56. Halaman Tambah Galeri Kegiatan.....	128
57. Halaman Ubah Galeri Kegiatan	130
58. Halaman Hapus Galeri Kegiatan.....	133
59. Kegiatan Pengujian Sistem	136

DAFTAR TABEL

Tabel	Halaman
1. Simbol-simbol <i>Use Case Diagram</i>	12
2. Simbol-simbol <i>Activity Diagram</i>	14
3. User Stories, Acceptance Test Criteria dan Value	20
4. Iteration Plan	23
5. Deskripsi <i>Use Case</i> untuk Melihat Visi Misi.....	74
6. Deskripsi <i>Use Case</i> untuk Melihat Struktur Organisasi	74
7. Deskripsi <i>Use Case</i> untuk Melihat Hymne.....	74
8. Deskripsi <i>Use Case</i> untuk Melihat Sejarah Singkat	74
9. Deskripsi <i>Use Case</i> untuk Melihat Prestasi	75
10. Deskripsi <i>Use Case</i> untuk Melihat Galeri Kegiatan	75
11. Deskripsi <i>Use Case</i> untuk Melihat Ekstrakurikuler.....	75
12. Deskripsi <i>Use Case</i> untuk Melihat <i>Ranking</i>	76
13. Deskripsi <i>Use Case</i> untuk Melihat <i>E-Raport</i>	76
14. Deskripsi <i>Use Case</i> untuk Menambah Prestasi.....	76
15. Deskripsi <i>Use Case</i> untuk Mengubah Prestasi	77
16. Deskripsi <i>Use Case</i> untuk Menghapus Prestasi	77
17. Deskripsi <i>Use Case</i> untuk Menambah Ekstrakurikuler	78
18. Deskripsi <i>Use Case</i> untuk Mengubah Ekstrakurikuler	78
19. Deskripsi <i>Use Case</i> untuk Menghapus Ekstrakurikuler	78
20. Deskripsi <i>Use Case</i> untuk Menambah Sejarah Singkat.....	79
21. Deskripsi <i>Use Case</i> untuk Mengubah Sejarah Singkat.....	79
22. Deskripsi <i>Use Case</i> untuk Menghapus Sejarah Singkat	80
23. Deskripsi <i>Use Case</i> untuk Menambah Galeri Kegiatan.....	80
24. Deskripsi <i>Use Case</i> untuk Mengubah Galeri Kegiatan	81

25. Deskripsi <i>Use Case</i> untuk Menghapus Galeri Kegiatan	81
26. Matriks Kebutuhan Fungsional As-Is dan To-Be	82
27. Matriks Kebutuhan Non-Fungsional As-Is dan To-Be	84
28. Skenario Black Box Testing.....	86
29. User Acceptance Testing Scenario.....	89
30. Hasil Pengujian Black Box	137
31. Skenario User Acceptance Testing	144
32. Hasil Jawaban Responden.....	144
33. Hasil Acceptance Testing.....	146

I. PENDAHULUAN

1.1 Latar Belakang

Sekolah sebagai suatu sistem diorganisasikan untuk memudahkan pencapaian tujuan belajar dan mengajar yang berkualitas dalam melayani peserta didik secara efektif dan efisien (Elyati et al., 2022). Dalam lingkungan sekolah SMPN 29 Bandar Lampung dengan jumlah siswa yang terus meningkat yakni memiliki 934 siswa, penyebaran informasi akademik dan kegiatan sekolah sering kali menjadi tantangan. Proses penyampaian informasi yang masih secara manual atau melalui media konvensional seringkali kurang efektif, memerlukan waktu yang lama, dan tidak terpusat. Akibatnya, siswa dan orang tua kesulitan dalam memperoleh informasi terkini mengenai akademik, kegiatan sekolah, serta evaluasi hasil belajar. Oleh karena itu, diperlukan sebuah sistem berbasis *web* agar dapat menyajikan informasi sekolah secara terpusat, transparan, dan mudah diakses.

Sekolah sebagai lembaga pendidikan dituntut untuk mampu menyediakan akses informasi secara cepat dan mudah. Namun, di SMPN 29 Bandar Lampung, sistem informasi akademik berbasis *website* belum tersedia, sehingga pengelolaan data masih dilakukan secara manual seperti mencatat di kertas atau menggunakan *spreadsheet* sederhana. Hal ini membuat rentan terjadinya kehilangan berkas, dan pengarsipannya di simpan dalam satu lemari buku, hal ini mempersulit pencarian data apabila sewaktu-waktu di perlukan (Ariansyah & Wijaya, 2021). Kesalahan – kesalahan seperti hilangnya dokumen, keterlambatan dalam distribusi informasi, dan sulitnya sinkronisasi data sering kali terjadi. Oleh karena itu, sistem informasi berbasis

website yang bisa diakses secara *online* melalui internet dapat menjadi solusi, sehingga guru, siswa, dan orang tua bisa dengan mudah melihat data akademik kapan saja dan di mana saja. Sistem ini tidak hanya membantu dalam penyampaian informasi, tetapi juga mendukung kegiatan evaluasi akademik secara lebih transparan dan efisien. Sistem informasi akademik ini menjadi sarana pendukung informasi akademik sehingga meningkatkan kualitas sesuai dengan kebutuhan (Adi Saputra et al., 2022).

SMPN 29 Bandar Lampung sebagai salah satu sekolah yang terus berkembang, membutuhkan sebuah sistem informasi akademik berbasis *website* yang dapat mengelola empat modul utama, yaitu modul informasi publik, modul data akademik, modul pembelajaran, dan modul evaluasi. Disini penulis bertanggungjawab atas modul informasi publik dan modul evaluasi. Dengan dirancangnya modul informasi publik yang menyediakan berbagai informasi tentang sekolah, seperti sejarah singkat, visi misi, hymne, struktur organisasi, ekstrakurikuler, prestasi, dan galeri kegiatan, penyebaran informasi menjadi lebih terpusat dan mudah diakses secara *online*. Hal ini memungkinkan aksesibilitas dan transparansi informasi yang lebih baik dan lebih cepat dalam menyampaikan kegiatan serta kebijakan sekolah, sekaligus memperkuat citra sekolah di mata masyarakat.

Dengan dirancangnya modul evaluasi yang dilengkapi dengan fitur seperti *ranking* dan *e-raport*, memungkinkan dapat mempermudah siswa dan orang tua dalam memantau hasil belajar secara *real-time online*. Fitur *ranking* akan memberikan transparansi yang lebih jelas mengenai perkembangan akademik mereka sehingga memotivasi siswa untuk menjadi lebih baik, berkompetisi dengan cara yang sehat, serta meningkatkan semangat belajar. Selain itu, modul evaluasi ini juga akan mempercepat proses pengolahan data akademik, mengurangi risiko kesalahan yang sering terjadi dalam evaluasi manual, dan meningkatkan efisiensi dalam penyajian data akademik siswa.

Penerapan metode *Extreme Programming* (XP) dalam pengembangan sistem ini dipilih karena pendekatannya yang fleksibel. XP merupakan metode yang memiliki tingkat responsivitas tinggi terhadap perubahan (Septiani & Habibie, 2022). XP menekankan kerja sama yang erat antara pengembang dan pengguna, serta memiliki siklus pengembangan yang cepat sehingga dapat menghadirkan sistem yang relevan dan sesuai dengan kebutuhan sekolah. *Framework* CodeIgniter 4 dipilih karena memudahkan pengembang *web* dalam membangun aplikasi *web* dengan cepat sejak awal. Selain membuat *website* lebih dinamis, penggunaan *framework* ini juga membantu pengembang untuk menciptakan aplikasi *web* yang ringan dan cepat (Ikhsan et al., 2023).

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, rumusan masalah yang dapat diajukan adalah sebagai berikut.

1. Bagaimana cara menyediakan akses informasi yang cepat, mudah, dan terkomputerisasi terkait SMPN 29 Bandar Lampung kepada masyarakat umum
2. Bagaimana penerapan sistem terkomputerisasi dapat mendukung peningkatan semangat belajar siswa dan transparansi hasil akademik di SMPN 29 Bandar Lampung

1.3 Batasan Masalah

Berdasarkan latar belakang yang telah diberikan, berikut adalah batasan masalah yang dapat ditetapkan untuk fokus penelitian.

1. Sistem informasi yang dikembangkan berbasis *website* dengan menggunakan *framework* CodeIgniter 4.
2. Sistem ini hanya digunakan di SMPN 29 Bandar Lampung.

3. Sistem ini dibangun menyesuaikan permintaan dan kebutuhan dari pihak SMPN 29 Bandar Lampung.

1.4 Tujuan Penelitian

Adapun tujuan dilakukannya penelitian ini adalah sebagai berikut.

1. Mengembangkan modul informasi publik yang memungkinkan penyediaan akses informasi yang cepat dan mudah terkait SMPN 29 Bandar Lampung, sehingga masyarakat umum dapat mengakses berbagai informasi penting tentang sekolah secara efisien dan transparan.
2. Mengembangkan modul evaluasi yang memungkinkan siswa dan orang tua untuk memantau hasil belajar secara *real-time* dan transparan, serta mengurangi kesalahan yang sering terjadi dalam evaluasi manual, sehingga data akademik dapat disajikan dengan lebih efisien.

1.5 Manfaat Penelitian

Adapun manfaat yang didapatkan dari penelitian ini adalah sebagai berikut.

1. Penelitian ini dapat membantu sekolah dalam menyampaikan informasi publik secara cepat, transparan, dan terstruktur. Modul informasi publik akan mempermudah akses terhadap informasi penting. Selain itu, modul evaluasi yang dilengkapi dengan fitur *ranking* dan *e-raport* akan memudahkan pihak sekolah dalam melakukan evaluasi akademik dengan lebih efisien dan akurat.
2. Penelitian ini juga bertujuan untuk meningkatkan kualitas pelayanan akademik kepada siswa dan orang tua melalui sistem yang mudah diakses dan memberikan informasi secara transparan.

II. TINJAUAN PUSAKA

2.1 Penelitian Terdahulu

Penelitian ini tidak terlepas dari penelitian-penelitian sebelumnya yang bertujuan untuk mendukung penelitian ini. Berikut ini merupakan skripsi dan jurnal penelitian yang digunakan sebagai referensi dalam penelitian.

2.1.1 Pengembangan Sistem Informasi Akademik berbasis *Web* di SMP Negeri 3 Sidikalang

Penelitian ini bertujuan untuk mengembangkan sistem informasi akademik berbasis *web* di SMP Negeri 3 Sidikalang yang mempermudah pengelolaan data akademik, seperti data siswa, guru, dan nilai, serta mempercepat penyebaran informasi. Penelitian ini menggunakan metode pengembangan Waterfall dengan tahapan pengumpulan kebutuhan, desain, implementasi, pengujian, dan pemeliharaan. Sistem dikembangkan menggunakan bahasa pemrograman PHP dengan *framework* CodeIgniter dan *database* MySQL. Pengujian sistem dilakukan menggunakan black-box *testing* untuk memastikan semua fungsi berjalan sesuai kebutuhan, serta *System Usability Scale* (SUS) untuk mengukur tingkat kegunaan sistem. Hasil pengujian menunjukkan skor rata-rata 77, yang masuk dalam kategori *acceptable* dengan grade C. Sistem ini dinyatakan layak digunakan sebagai alat bantu pengelolaan data akademik dan penyebaran informasi di lingkungan SMP Negeri 3 Sidikalang (Sahat et al., 2023).

2.1.2 Penerapan *Extreme Programming* Dalam Sistem Informasi Akademik SDN Kuala Teladas

Penelitian ini bertujuan untuk merancang dan membangun sistem informasi akademik berbasis *website* di SD Negeri Kuala Teladas dengan metode pengembangan *Extreme Programming* (XP). Sistem ini dirancang untuk mempermudah pengolahan dan penyajian data akademik seperti nilai siswa dan data administrasi sekolah. Penelitian menggunakan bahasa pemrograman PHP dan SQLyog untuk pengelolaan *database*. Pengujian sistem dilakukan menggunakan *Technology Acceptance Model* (TAM), dengan hasil rata-rata tingkat kelayakan sebesar 92,71%, yang termasuk dalam kategori sangat layak. Sistem ini dinilai mampu meningkatkan efektivitas pengelolaan data akademik sekolah secara mudah, cepat, dan dapat diakses kapan saja tanpa harus datang langsung ke sekolah. Penelitian ini menunjukkan bahwa metode XP yang fleksibel dan adaptif menjadi solusi efektif untuk memenuhi kebutuhan pengguna dalam membangun sistem informasi akademik di lingkungan SD Negeri Kuala Teladas (Ardiansah et al., 2023).

2.1.3 Sistem Informasi Akademik Berbasis *Website* (Studi Kasus: SMPIT AVICENNA)

Penelitian ini bertujuan untuk mengembangkan sistem informasi akademik berbasis *web* di SMPIT Avicenna yang mampu mengatasi kelemahan sistem konvensional, seperti pencatatan manual dan penggunaan media kertas, yang sering kali memakan waktu dan rentan terhadap kesalahan. Sistem ini dirancang menggunakan metode *System Development Life Cycle* (SDLC) dengan model Waterfall, melalui tahapan analisis, desain menggunakan UML, pengkodean dengan PHP, dan pengujian menggunakan metode *Black Box Testing*. Sistem informasi akademik ini melibatkan enam aktor, yaitu admin, tata usaha, kepala sekolah, guru, siswa, dan kurikulum. Implementasinya mencakup fitur untuk pengelolaan nilai, pembayaran SPP, dan

penyediaan informasi akademik lainnya. Pengujian menunjukkan bahwa sistem ini efektif dalam meningkatkan efisiensi proses akademik, seperti pencatatan nilai dan pembayaran SPP secara daring, serta mempermudah akses informasi bagi siswa dan orang tua. Penelitian ini menunjukkan bahwa sistem berbasis *web* dapat mengatasi tantangan pengelolaan data akademik secara efisien dan akurat (Hakim & Meilina, 2022).

2.1.4 Model Pengembangan Sistem Informasi Akademik Berbasis User Centered Design Menerapkan Framework Flask Python

Penelitian ini bertujuan mengembangkan sistem informasi akademik berbasis *web* di SMK Negeri 1 Bitung untuk menggantikan sistem manual yang kurang efektif. Sistem ini dirancang menggunakan metode *User Centered Design* (UCD) dan dikembangkan dengan *framework* Flask Python melalui tahapan perancangan konsep, pengumpulan data, perancangan aplikasi, dan implementasi. Sistem melibatkan beberapa aktor seperti admin, guru, dan siswa, dengan fitur utama meliputi pengelolaan data siswa, guru, jadwal pelajaran, nilai, dan pengumuman. Pengujian menggunakan metode *Black Box Testing* menunjukkan sistem mampu meningkatkan efisiensi proses akademik dengan fitur CRUD yang berjalan baik dan memudahkan akses data secara daring. Penelitian ini membuktikan bahwa penerapan sistem berbasis *web* dengan Flask dapat meningkatkan kecepatan, akurasi, dan kemudahan pengelolaan informasi akademik (Ngantung & Pakereng, 2021).

2.1.5 Pengembangan Sistem Informasi Akademik Berbasis Mobile Menggunakan Flutter di Universitas Narotama Surabaya

Penelitian ini bertujuan mengembangkan aplikasi Sistem Informasi Akademik berbasis *mobile* di Universitas Narotama Surabaya menggunakan *framework* Flutter dan bahasa pemrograman Dart. Sistem ini dibuat untuk mengatasi keterbatasan akses sistem berbasis *web* yang hanya nyaman digunakan di perangkat komputer, dengan

memanfaatkan tren penggunaan perangkat *mobile* yang jauh lebih tinggi. Metode pengembangan menggunakan model Waterfall yang meliputi analisis kebutuhan, desain sistem, implementasi, integrasi dan pengujian, serta pemeliharaan. Fitur utama aplikasi meliputi pengelolaan data pribadi, KRS, jadwal kuliah, bimbingan online, KHS, dan transkrip nilai. Sistem ini memudahkan mahasiswa mengakses informasi akademik secara *mobile* dengan antarmuka yang responsif dan *user-friendly*. Penelitian ini menunjukkan aplikasi berbasis Flutter efektif mendukung kegiatan akademik dengan kemudahan akses dan performa yang baik, meskipun pengembangan masih difokuskan pada *platform* Android dan perlu pengembangan lanjutan untuk *platform* iOS (Pratama & Kamisutara, 2021).

2.1.6 Sistem Informasi Akademik Santri Berbasis Web di Pondok Pesantren

Penelitian ini bertujuan merancang sistem informasi akademik santri berbasis *web* di Pondok Pesantren Al Halim Garut untuk memudahkan pengelolaan data akademik dan pembayaran secara *online*. Metode pengembangan yang digunakan adalah *Rational Unified Process* (RUP) yang terdiri dari empat tahapan utama: *inception* (observasi, wawancara, identifikasi literatur, perancangan proses bisnis, dan spesifikasi sistem), *elaboration* (analisis dan desain sistem menggunakan *Unified Modeling Language* seperti *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*), *construction* (implementasi kode program), dan *transition* (pengujian aplikasi). Pengujian dilakukan dengan *Black Box Testing*, yaitu pengujian fungsional berdasarkan spesifikasi tanpa melihat kode sumber, menggunakan teknik *State Transition Testing* untuk memastikan semua fungsi aplikasi berjalan sesuai kebutuhan. Hasil penelitian menunjukkan sistem dapat memudahkan pengelolaan akademik santri

dan pembayaran secara daring serta meningkatkan efisiensi di pondok pesantren tersebut (Sutedi et al., 2021).

2.2 Uraian Landasan Teori

Beberapa teori yang berkaitan dengan penelitian ini adalah sebagai berikut.

2.2.1 Pengertian Sistem Informasi Akademik

Menurut Rakian dkk. (2022), dalam jurnal berjudul “Perancangan Sistem Informasi Akademik Berbasis *Web* Pada SMK” menyatakan bahwa Sistem informasi akademik adalah sebuah rangkaian aktifitas yang mendiskripsikan alur data dapat terhubung dan diproses menjadi informasi yang akan dikonsumsi oleh pengguna (Rakian et al., 2022). Menurut Solahudin (2021), Sistem Informasi Akademik adalah salah-satu aplikasi yang dirancang untuk kebutuhan pengolahan data administratif sekolah dengan tujuan supaya data akademik lebih terkelola dengan baik dalam pengambilan keputusan di lingkungan sekolah (Solahudin, 2021).

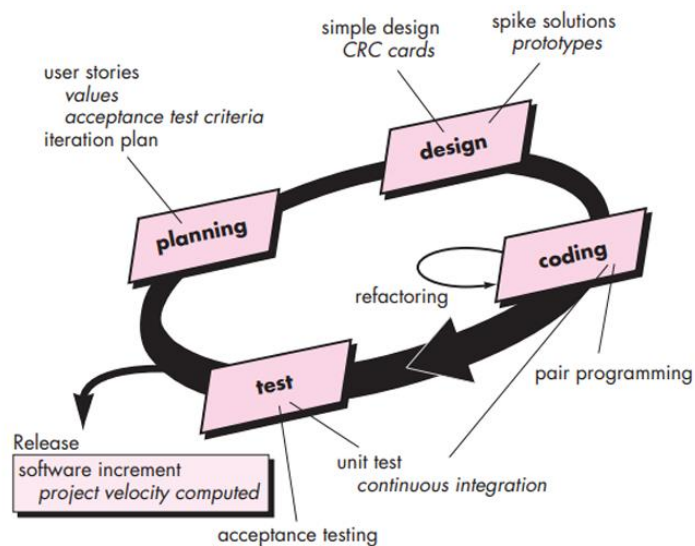
2.2.2 Pengertian PHP

Burhanuddin Damanik menyatakan dalam “*Rancangan Sistem Informasi Smp Negeri 1 Tuhemberua Kabupaten Nias Utara Menggunakan Php Codeigniter*” (2021) bahwa PHP merupakan kependekan dari PHP Hypertext Preprocessor yaitu bahasa pemrograman *web* yang dapat disisipkan dalam skrip HTML dan bekerja di sisi server. Tujuan dari bahasa ini adalah membantu para pengembangan *web* untuk membuat *web* dinamis dengan cepat. Agar dapat menjalankan PHP harus menyediakan perangkat lunak berikut *web* server (Apache, IIS, personal web server/PWS), PHP server dan database Server (MySQL, Iterbase, MS SQL) (Damanik, 2021). Putra dkk. (2023) menyatakan bahwa PHP adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan

pengembangan sebuah *web* dan bisa digunakan pada HTML (Putra et al., 2023).

2.2.3 Pengertian *Extreme Programming* (XP)

Menurut J Coyanda dkk. (2022), dalam jurnal berjudul “*Sistem Informasi Ekspedisi Barang Pada PT. New Power Global Energy Dengan Menggunakan Metode Extreme Programming*” menyatakan bahwa Extreme Programming (XP) adalah metode pengembangan software yang cepat, efisien, beresiko rendah, fleksibel, terprediksi, scientific, dan menyenangkan karena penilaian kinerja sistem diperlukan dalam waktu yang cepat sementara ada kebutuhan sistem yang masih berubah, Model ini cenderung menggunakan pendekatan *Object-Oriented*. Tahapan-tahapan yang harus dilalui antara lain: *Planning*, *Design*, *Coding*, dan *Testing* (Coyanda et al., 2022). Adapun alur tahapan *Extreme Programming* dapat dilihat pada Gambar 1.



Gambar 1. Alur Tahapan Metode Extreme Programming.

(Kustiawan et al., 2022)

Berikut adalah penjelasan mengenai keempat tahapan diatas:

1) *Planning* (Perencanaan)

Tahapan ini merupakan langkah awal dalam pembangunan sistem dimana dalam tahapan ini dilakukan beberapa kegiatan perencanaan yaitu, identifikasi permasalahan, menganalisa kebutuhan sampai dengan penetapan jadwal pelaksanaan pembangunan sistem.

2) *Design (Perancangan)*

Tahapan berikutnya adalah perancangan dimana pada tahapan ini dilakukan kegiatan pemodelan yang dimulai dari pemodelan sistem, pemodelan arsitektur sampai dengan pemodelan basis data.

3) *Coding (Pengkodean)*

Tahapan ini merupakan kegiatan penerapan pemodelan yang sudah dibuat kedalam bentuk *user interface* dengan menggunakan bahasa pemrograman.

4) *Testing (Pengujian)*

Setelah tahapan pengkodean selesai, kemudian dilakukan tahapan pengujian sistem untuk mengetahui kesalahan apa saja yang timbul saat aplikasi sedang berjalan serta mengetahui apakah sistem yang dibangun sudah sesuai dengan kebutuhan pengguna. dan penekanan pada jawaban atas permasalahan. (Coyanda et al., 2022)

2.2.4 Pengertian *Unified Modeling Language (UML)*

Dalam bukunya, Rumbaugh dkk. (2021) menyatakan bahwa *Unified Modeling Language (UML)* adalah bahasa pemodelan visual serbaguna yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan artefak dari sebuah sistem perangkat lunak. Bahasa ini menangkap keputusan dan pemahaman tentang sistem yang harus dibangun. UML digunakan untuk memahami, merancang, menjelajah, mengonfigurasi, memelihara, dan mengendalikan informasi tentang sistem tersebut. Bahasa ini ditujukan untuk digunakan dengan berbagai metode pengembangan, tahapan siklus hidup, domain aplikasi, dan media (Rumbaugh et al., 2021). UML menyediakan berbagai jenis diagram untuk memodelkan aspek

struktural dan perilaku sistem, seperti *Use case diagram* dan *Activity diagram*. Dengan menggunakan UML, pengembang dapat memvisualisasikan, menentukan, dan mendokumentasikan desain sistem secara sistematis, sehingga mempermudah komunikasi antara tim pengembang, pemangku kepentingan, dan pengguna. UML juga memungkinkan identifikasi kebutuhan sistem secara lebih terstruktur dan jelas sebelum proses pengembangan dimulai.

1. *Use Case Diagram*

Use case menggambarkan perilaku suatu sistem, subsistem, atau kelas sebagaimana terlihat oleh pengguna luar. Ini membagi fungsi sistem menjadi transaksi yang bermakna bagi aktor—pengguna ideal dari sistem tersebut. Bagian dari fungsi interaktif ini disebut *use case*. *Use case* mendeskripsikan interaksi dengan aktor sebagai rangkaian pesan antara sistem dan satu atau lebih aktor. Istilah aktor mencakup manusia, serta sistem dan proses komputer lainnya (Rumbaugh et al., 2021). Adapun simbol-simbol *Use Case Diagram* disajikan pada Tabel 1.

Tabel 1. Simbol-simbol *Use Case Diagram*

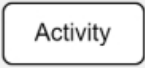
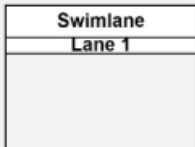
No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Aktor merepresentasikan entitas luar yang berinteraksi dengan sistem. Aktor bisa berupa: Manusia, Sistem Eksternal, Perangkat Keras.
2		<i>Association</i>	<i>Association</i> merupakan hubungan yang menunjukkan interaksi antara aktor dengan <i>use case</i> .
3		<i>Use Case</i>	<i>Use Case</i> menggambarkan fungsi atau aktivitas yang

			dilakukan oleh sistem untuk memenuhi kebutuhan aktor.
4		Relasi <i>Include</i>	Suatu relasi yang menghubungkan suatu <i>use case</i> dengan <i>use case</i> tambahan, dimana <i>use case</i> tambahan tersebut tidak dapat menjalankan fungsinya tanpa <i>use case</i> utama
5		Relasi <i>Extend</i>	Relasi yang menghubungkan satu <i>use case</i> dengan <i>use case</i> tambahan, dimana <i>use case</i> tambahan tersebut dapat menjalankan fungsi meskipun tidak ada <i>use case</i> utama.

2. Activity Diagram

Activity diagram adalah notasi untuk grafik aktivitas. Diagram ini mencakup beberapa simbol singkatan khusus untuk kemudahan. Simbol-simbol ini sebenarnya bisa digunakan pada diagram statechart mana pun, meskipun pencampuran notasi mungkin terlihat kurang bagus sebagian besar waktu. *Activity state* ditampilkan sebagai kotak dengan ujung melengkung yang berisi deskripsi aktivitas. Transisi penyelesaian sederhana ditunjukkan dengan panah. Cabang-cabang ditunjukkan sebagai kondisi pengawal pada transisi atau sebagai berlian dengan beberapa panah keluar berlabel. *Fork* atau join kontrol ditampilkan sama seperti pada *statechart*, dengan beberapa panah yang masuk atau keluar melewati batang sinkronisasi yang berat (Rumbaugh et al., 2021). Adapun simbol-simbol *Activity diagram* disajikan pada Tabel 2.

Tabel 2. Simbol-simbol *Activity Diagram*

No	Gambar	Nama	Keterangan
1		<i>Initial State</i>	Indikasi awal dimulainya operasi yang dilakukan oleh sistem.
2		<i>Finale Node</i>	Indikasi akhir penyelesaian operasi yang dilakukan oleh sistem.
3		<i>Activity</i>	Simbol aktivitas yang dilakukan dalam sistem dan dimulai dengan kata kerja.
4		<i>Decision</i>	Simbol cabang dimana terdapat lebih dari satu pilihan.
5		<i>Swimlane</i>	Diagram alir yang menunjukkan objek melakukan aktivitas.

2.2.5. *Class Diagram*

Class diagram menggambarkan hubungan antar kelas beserta penjelasan detail masing-masing kelas dalam model desain suatu sistem. Diagram ini juga menampilkan aturan serta tanggung jawab entitas yang menentukan perilaku sistem. Dengan demikian, *class diagram* dapat diartikan sebagai representasi visual dari struktur sistem program berdasarkan jenis-jenis yang terbentuk. *Class diagram* menunjukkan alur kerja *database* dalam sistem yang akan dikembangkan. Selain itu, *class diagram* merupakan kumpulan

beberapa kelas beserta relasinya. Kelas identik dengan entitas yang digambarkan dalam bentuk persegi dengan nama kelas pada bagian atas, diikuti oleh atribut di bagian tengah, dan metode di bagian bawah (Ramdany et al., 2024). *Class diagram* merupakan salah satu pemodelan yang sangat penting dalam UML, yang berfungsi untuk membangun model logis dari sebuah sistem. Diagram ini menggambarkan skema arsitektur sistem yang sedang dikembangkan. *Class diagram* terdiri dari kelas-kelas yang memuat atribut dan metode, dimana setiap kelas saling terhubung melalui garis yang disebut asosiasi (Aliman, 2021).

2.2.6. Pengertian *CodeIgniter 4*

Menurut Dristyan dkk. (2021) pada jurnal “Sistem Informasi Geografis Penyebaran Covid-19 Di Kabupaten Asahan Menggunakan *Framework Codeigniter 4*”, disebutkan bahwa *CodeIgniter 4* adalah sebuah *framework* PHP sumber terbuka yang menggunakan pola arsitektur *Model-View-Controller* (MVC) untuk membangun aplikasi *web* dinamis. *Framework* ini memudahkan pengembang dalam membangun situs *web* secara cepat dan efisien, tanpa harus membuat aplikasi dari awal. Pada pola MVC, bagian *View* berfungsi untuk menangani logika presentasi, yakni menerima dan menampilkan data kepada pengguna. *Model* bertanggung jawab untuk berinteraksi dengan basis data, melakukan operasi seperti penyisipan, pembaruan, penghapusan, dan pencarian data. Sementara itu, *Controller* berperan dalam mengatur alur kerja antara *Model* dan *View*, menerima permintaan serta data dari pengguna, dan menentukan proses yang akan dijalankan oleh aplikasi (Dristyan et al., 2021).

2.2.7. Pengertian *User Acceptance Testing* (UAT)

Menurut Eko Suprpto (2021) pada jurnal “*User Acceptance Testing* (UAT) *Refreshment* PBX Outlet Site BNI Kanwil Padang”, disebutkan bahwa *User Acceptance Testing* (UAT) merupakan proses verifikasi

bahwa solusi yang dibuat dalam sistem sudah sesuai untuk pengguna. Proses ini berbeda dengan pengujian sistem (memastikan *software* tidak *crash* dan sesuai dengan dokumen permintaan pengguna), melainkan memastikan bahwa solusi dalam sistem tersebut akan bekerja untuk pengguna (yaitu, tes bahwa pengguna menerima solusi di dalam sistem). UAT umumnya dilakukan oleh klien atau pengguna akhir, biasanya tidak fokus pada identifikasi masalah sederhana seperti kesalahan ejaan, maupun di cacat *showstopper*, seperti *crash* perangkat lunak. Penguji dan pengembang mengidentifikasi dan memperbaiki masalah ini selama tahap awal pengujian fungsionalitas, pengujian saat integrasi dan pada tahap sistem *testing* (Suprpto, 2021).

Berikut ini adalah rumus perhitungan presentase berdasarkan skor dari skala yang sudah ditentukan

$$P = \frac{S}{\text{Skor}} \times 100\% \dots (\text{Chamida et al., 2021})$$

Keterangan:

P = Nilai presentase

S = Jumlah frekuensi dikalikan dengan skor jawaban

Skor = Skor tertinggi dikali dengan Jumlah ideal sampel

2.2.8. Pengertian Black Box Testing

Metode Blackbox *Testing* adalah teknik pengujian perangkat lunak yang dilakukan tanpa melihat struktur atau kode program secara detail. Pengujian ini dilakukan dengan memasukkan data ke dalam setiap form yang ada pada aplikasi untuk mengevaluasi bagaimana program merespons input tersebut. Tujuan pengujian ini adalah memastikan bahwa perangkat lunak berjalan sesuai dengan kebutuhan dan harapan perusahaan (Made et al., 2021).

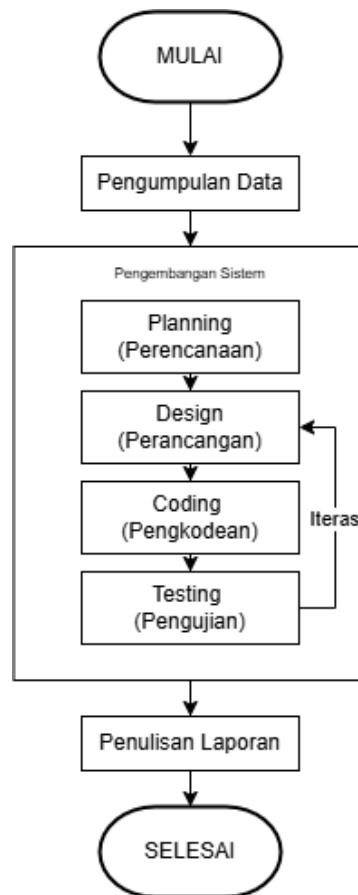
III. METODOLOGI PENELITIAN

3.1. Waktu dan Tempat Pelaksanaan

Penelitian ini dilakukan pada semester ganjil tahun akademik 2024/2025 di Gedung Ilmu Komputer, Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung, serta di SMPN 29 Bandar Lampung yang berlokasi di Jl. Penghijauan No.50, Way Dadi, Kec. Sukarame, Kota Bandar Lampung.

3.2. Tahapan Penelitian

Proses penelitian terdiri dari tiga bagian utama yang saling terkait, yaitu pengumpulan data, pengembangan sistem, dan penulisan laporan. Tahap pengumpulan data diperoleh melalui wawancara, studi pustaka, dan observasi langsung untuk memahami alur kerja bisnis serta teori yang akan diterapkan dalam penelitian. Selanjutnya, pada tahap pengembangan sistem, data dan informasi tersebut digunakan untuk merancang, membuat, dan menguji sistem sesuai kebutuhan. Proses ini mencakup pembuatan rancangan sistem, penulisan program, serta pengujian agar sistem berjalan dengan baik. Terakhir, tahap penulisan laporan, bertujuan untuk menyajikan hasil penelitian secara lengkap dan mendokumentasikan seluruh proses yang telah dilakukan, mulai dari latar belakang, teori, metode, hasil, hingga kesimpulan dan saran. Tahapan proses penelitian dapat dilihat pada Gambar 2.



Gambar 2. Tahapan Penelitian

3.2.1. Tahap Pengumpulan Data

Tahap pengumpulan data diperoleh melalui wawancara, studi pustaka, dan observasi. Pengumpulan data dilakukan untuk memahami proses alur kerja bisnis yang terjadi di SMPN 29 Bandar Lampung serta untuk memahami teori-teori yang akan diterapkan dalam penelitian ini. Tahapan pengumpulan data adalah sebagai berikut:

1. Wawancara

Metode wawancara digunakan untuk mengumpulkan informasi dari pihak-pihak terkait di SMPN 29 Bandar Lampung, seperti kepala sekolah, guru, dan staf tata usaha. Proses wawancara ini bertujuan untuk mendapatkan informasi tentang sekolah, seperti sejarah, visi

dan misi, struktur organisasi, hymne, serta kegiatan ekstrakurikuler. Selain itu, wawancara juga menggali kebutuhan sekolah terkait sistem evaluasi baru yang diinginkan, yaitu melalui penerapan *e-ranking* dan *e-raport*. Informasi yang diperoleh berupa data tertulis maupun lisan, yang menjadi dasar dalam merancang dan mengembangkan modul informasi publik dan evaluasi yang sesuai dengan kebutuhan sekolah.

2. Studi Pustaka

Metode studi pustaka dilakukan dengan mencari informasi dari berbagai sumber seperti buku, jurnal, artikel ilmiah, dan skripsi yang dijadikan referensi dalam penyusunan laporan penelitian. Sumber-sumber tersebut berkaitan dengan topik yang diteliti, yaitu pengembangan sistem informasi akademik. Studi pustaka ini bertujuan untuk memperoleh landasan teori yang relevan serta memperkaya pemahaman peneliti terkait konsep sistem informasi, manajemen data akademik, dan teknologi yang digunakan dalam pengembangan sistem.

3. Observasi

Observasi dilakukan dengan mengamati secara langsung kondisi dan proses yang berjalan di sekolah terkait penyajian data seperti sejarah sekolah, visi dan misi, struktur organisasi, hymne, galeri kegiatan, prestasi, dan ekstrakurikuler sekolah yang saat ini masih dilakukan tanpa sistem dan belum dapat diakses secara luas oleh semua pihak. Selain itu, observasi juga dilakukan untuk melihat bagaimana proses evaluasi saat ini berjalan, terutama terkait penilaian siswa tanpa menggunakan sistem *e-ranking* dan *e-raport*. Tujuannya adalah untuk mengetahui kondisi nyata dan kebutuhan yang harus dipenuhi dalam pengembangan modul informasi publik dan evaluasi yang akan dibuat.

3.2.2. Tahap Pengembangan Sistem

Tahap selanjutnya adalah pengembangan sistem dengan menggunakan metode *Extreme Programming* (XP). Tahapan yang dilakukan dalam metode *Extreme Programming* meliputi perencanaan, perancangan, pengkodean, dan pengujian.

1. *Planning* (Perencanaan)

A. *User Stories, Acceptance Test Criteria* dan *Value*

User stories merupakan deskripsi singkat mengenai kebutuhan atau keinginan pengguna sistem, yang dalam konteks ini meliputi siswa, guru, dan admin. Untuk memastikan *user story* tersebut dapat dianggap selesai dan berhasil, digunakan *acceptance test criteria*, yaitu kriteria yang harus dipenuhi oleh sistem. Selain itu, terdapat nilai atau *value* yang menunjukkan prioritas, atau pentingnya setiap *user story* dalam pengembangan sistem yang mana berada pada jarak 1 yang paling rendah hingga 10 yang paling tinggi.

Tabel 3. *User Stories, Acceptance Test Criteria* dan *Value*

No	<i>User Stories</i>	<i>Acceptance Test Criteria</i>	<i>Value</i>
1	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat <i>ranking</i> siswa di dalam sistem, agar saya bisa mengetahui posisi saya atau siswa lain berdasarkan hasil akademik.	Sistem menampilkan <i>ranking</i> siswa secara akurat berdasarkan hasil akademik terbaru sehingga pengguna dapat mengetahui posisi siswa dengan jelas dan <i>real-time</i> .	9
2	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat dan mencetak <i>E-Raport</i> saya atau siswa di dalam sistem, agar saya bisa	Sistem menyediakan fitur untuk melihat dan mencetak <i>E-Raport</i> secara digital dan <i>real-time</i> , sehingga pengguna dapat mengakses laporan hasil	9

	mendapatkan ringkasan laporan hasil belajar secara digital dan <i>realtime</i> .	belajar kapan saja dan mencetaknya dengan mudah.	
3	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat visi dan misi sekolah di dalam sistem, agar saya bisa memahami tujuan dan arah yang ingin dicapai oleh sekolah.	Sistem menampilkan visi dan misi sekolah secara jelas dan mudah diakses oleh semua pengguna sehingga dapat dipahami dengan baik.	8
4	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat struktur organisasi sekolah di dalam sistem, agar saya dapat mengetahui siapa saja yang terlibat dalam pengelolaan sekolah dan tugas masing-masing.	Sistem menyediakan informasi struktur organisasi sekolah yang lengkap dan terbaru sehingga pengguna dapat mengetahui peran dan tugas masing-masing anggota organisasi dengan jelas.	8
5	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat lirik dan memutar video hymne sekolah melalui sistem, agar saya dapat lebih mengenal nilai-nilai yang terkandung dalam hymne sekolah.	Sistem memungkinkan pengguna untuk melihat lirik dan memutar video hymne sekolah dengan kualitas baik sehingga pengguna dapat memahami nilai-nilai dalam hymne tersebut.	8
6	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat sejarah singkat sekolah di dalam sistem, agar saya dapat mengetahui latar belakang dan perjalanan sekolah hingga saat ini.	Sistem menampilkan sejarah singkat sekolah secara informatif dan mudah dipahami sehingga pengguna dapat mengetahui perkembangan sekolah dari masa ke masa.	7
7	Sebagai pengguna (Admin), saya ingin dapat mengelola sejarah singkat sekolah (menambah, mengubah, atau menghapus) di dalam sistem, agar	Sistem memberikan hak akses kepada admin untuk menambah, mengubah, dan menghapus data sejarah sekolah sehingga data tetap akurat dan terupdate.	7

	saya bisa menjaga keakuratan informasi sejarah kepemimpinan sekolah.		
8	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat daftar prestasi yang telah diraih oleh siswa dan sekolah melalui sistem, agar saya bisa mendapatkan informasi mengenai pencapaian-pencapaian yang membanggakan.	Sistem menampilkan daftar prestasi siswa dan sekolah secara lengkap dan terupdate sehingga pengguna dapat melihat pencapaian-pencapaian yang ada dengan jelas.	7
9	Sebagai pengguna (Admin), saya ingin dapat mengelola prestasi (menambah, mengubah, atau menghapus) di dalam sistem, agar saya bisa memperbarui dan memelihara data prestasi siswa.	Sistem memungkinkan admin untuk mengelola data prestasi siswa, termasuk menambah, mengubah, dan menghapus informasi agar data tetap akurat dan terbaru.	7
10	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat daftar ekstrakurikuler yang tersedia di sekolah melalui sistem, agar saya bisa memilih dan mendaftar kegiatan ekstrakurikuler yang sesuai dengan minat saya.	Sistem menampilkan daftar ekstrakurikuler yang lengkap sehingga siswa dapat memilih dan mendaftar kegiatan sesuai dengan minat mereka.	7
11	Sebagai pengguna (Admin), saya ingin dapat mengelola ekstrakurikuler (menambah, mengubah, atau menghapus) di dalam sistem, agar saya bisa memperbarui kegiatan ekstrakurikuler yang relevan.	Sistem memberikan hak akses kepada admin untuk mengelola data ekstrakurikuler, termasuk menambah, mengubah, dan menghapus kegiatan sesuai kebutuhan sekolah.	7

12	Sebagai pengguna (Siswa, Guru, Admin), saya ingin dapat melihat galeri foto kegiatan sekolah di dalam sistem, agar saya bisa melihat momen-momen penting dan kegiatan yang telah dilaksanakan di sekolah.	Sistem menyediakan galeri foto kegiatan sekolah yang mudah diakses dan menampilkan dokumentasi kegiatan secara menarik sehingga pengguna dapat melihat aktivitas sekolah dengan jelas.	6
13	Sebagai pengguna (Admin), saya ingin dapat mengelola galeri kegiatan (menambah, mengubah, atau menghapus) di dalam sistem, agar saya bisa memastikan semua kegiatan penting tercatat dengan baik.	Sistem memungkinkan admin untuk mengelola galeri kegiatan, termasuk menambah, mengubah, dan menghapus foto atau album kegiatan agar dokumentasi sekolah selalu terorganisir dengan baik.	6

B. Iteration Plan

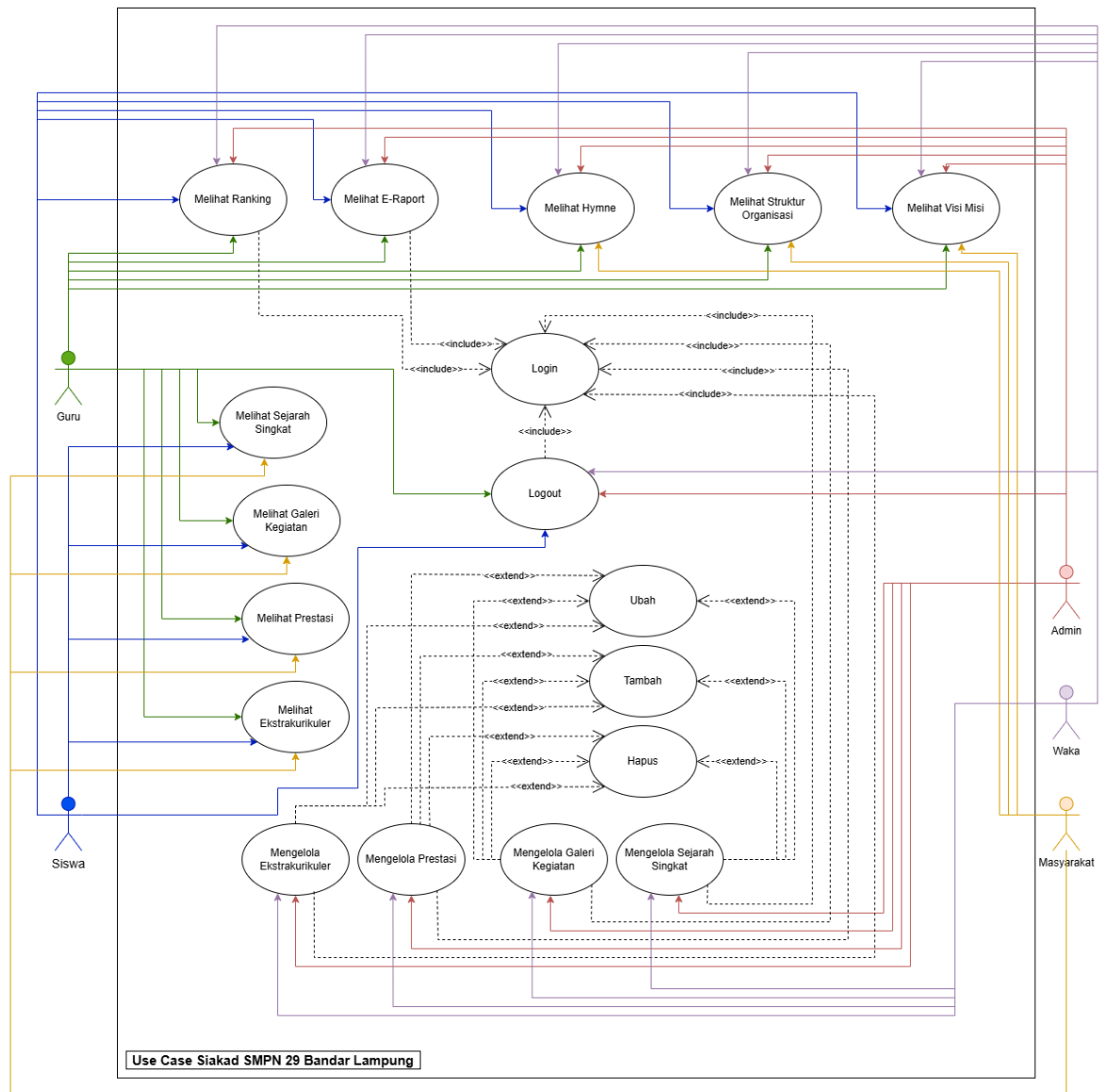
Pada tahap ini, rencana iterasi ditentukan berdasarkan *user story* yang telah dibuat sebelumnya dan telah diurutkan berdasarkan tingkat kepentingan mulai dari 1 paling rendah hingga 10 paling tinggi. Berikut adalah rencana iterasi yang dapat dilihat pada Tabel 4.

Tabel 4. Iteration Plan

Iterasi	Fungsional	Kriteria Penerimaan	Nilai
1.	Dapat melihat data <i>ranking</i> siswa	Data <i>Ranking</i> siswa berhasil ditampilkan	9
	Dapat melihat <i>E-Raport</i> siswa	<i>E-Raport</i> berhasil ditampilkan	9
	Dapat mencetak <i>E-Raport</i> siswa	<i>E-Raport</i> berhasil dicetak	9
	Dapat melihat visi dan misi sekolah	Halaman visi dan misi berhasil ditampilkan	8
	Dapat melihat stuktur organisasi sekolah	Halaman struktur organisasi berhasil ditampilkan	8

	Dapat melihat lirik serta memutar video hymne sekolah	Lirik hymne berhasil tampil dan video dapat diputar	8
	Dapat melihat sejarah singkat sekolah	Halaman sejarah singkat berhasil ditampilkan	7
	Dapat menambah sejarah kepemimpinan sekolah	Sejarah kepemimpinan berhasil ditambahkan	7
	Dapat mengubah sejarah kepemimpinan sekolah	Sejarah kepemimpinan berhasil diubah	7
	Dapat menghapus sejarah kepemimpinan sekolah	Sejarah kepemimpinan berhasil dihapus	7
	Dapat melihat daftar prestasi sekolah	Halaman data prestasi berhasil ditampilkan	7
	Dapat menambah data prestasi sekolah	Data prestasi berhasil ditambahkan	7
	Dapat mengubah data prestasi sekolah	Data prestasi berhasil diubah	7
	Dapat menghapus data prestasi sekolah	Data prestasi berhasil dihapus	7
	Dapat melihat daftar ekstrakurikuler sekolah	Data ekstrakurikuler berhasil ditampilkan	7
	Dapat menambah data ekstrakurikuler	Data ekstrakurikuler berhasil ditambahkan	7
	Dapat mengubah data ekstrakurikuler	Data ekstrakurikuler berhasil diubah	7
	Dapat menghapus data ekstrakurikuler	Data ekstrakurikuler berhasil dihapus	7
	Dapat melihat galeri foto kegiatan sekolah	Galeri kegiatan berhasil ditampilkan	6
	Dapat menambah data galeri kegiatan	Data galeri kegiatan berhasil ditambahkan	6
	Dapat mengubah data galeri kegiatan	Data galeri kegiatan berhasil diubah	6
	Dapat menghapus data galeri kegiatan	Data galeri kegiatan berhasil dihapus	6

C. Use Case Diagram



Gambar 3. Use Case Diagram

Use case diagram Sistem Informasi Akademik SMPN 29 Bandar Lampung menggambarkan interaksi antara lima aktor, yaitu Guru, Siswa, Admin, Waka dan masyarakat dengan berbagai fungsi yang tersedia dalam sistem. Semua aktor harus melakukan *login* terlebih dahulu untuk mengakses beberapa fitur yang disediakan dan dapat melakukan *logout* setelah selesai menggunakan sistem, namun terdapat pula beberapa fitur yang

langsung bisa diakses masyarakat tanpa melakukan *login*. Guru dan Siswa dapat mengakses informasi penting seperti melihat *ranking* siswa, melihat *E-Raport*, serta melihat konten sekolah seperti hymne, struktur organisasi, visi dan misi, sejarah singkat, galeri kegiatan, prestasi, dan ekstrakurikuler. Sementara itu, Admin dan Waka memiliki peran khusus untuk mengelola data-data tersebut, termasuk kemampuan menambah, mengubah, dan menghapus informasi pada modul ekstrakurikuler, prestasi, galeri kegiatan, dan sejarah singkat. Proses pengelolaan data ini meliputi fungsi-fungsi seperti tambah, ubah, dan hapus yang menjadi bagian dari aktivitas pengelolaan utama. Dengan struktur ini, sistem memastikan hak akses pengguna sesuai peran masing-masing, menjaga keamanan melalui autentikasi, serta menyediakan kemudahan dalam pengelolaan dan penyajian informasi akademik secara online bagi seluruh pemangku kepentingan. Penjabaran *Use Case* dapat dilihat pada sub bab *Use Case Description* dan *Activity Diagram* pada tahapan *Design*.

D. Ringkasan Eksekutif

Pada proyek ini, dikembangkan sebuah sistem informasi berbasis *website* dengan fokus pada modul Informasi Publik dan modul Evaluasi untuk mendukung kebutuhan akademik di SMPN 29 Bandar Lampung. Sistem ini dirancang untuk memberikan kemudahan kepada pengguna, seperti siswa, guru, dan orang tua, dalam mengakses informasi akademik dan memantau perkembangan hasil belajar secara digital. Modul Informasi Publik menyediakan berbagai informasi penting, seperti sejarah sekolah, visi dan misi, struktur organisasi, kegiatan ekstrakurikuler, prestasi, dan galeri kegiatan sekolah. Modul ini diharapkan dapat menjadi sarana transparansi informasi bagi masyarakat umum dan *stakeholder* sekolah. Modul Evaluasi, di

sisi lain, memungkinkan pengelolaan data evaluasi siswa, termasuk *e-Rapor* dan peringkat siswa. Modul ini dirancang untuk mempermudah guru dalam proses *input* data evaluasi, serta memberikan akses yang cepat dan akurat kepada siswa dan orang tua dalam memantau hasil belajar.

Proyek ini menggunakan *framework* CodeIgniter 4 yang dipilih karena kemampuannya dalam membangun aplikasi *web* yang ringan, cepat, dan dinamis. Selain itu, metodologi pengembangan perangkat lunak yang digunakan adalah *Extreme Programming* (XP), yang sangat responsif terhadap perubahan kebutuhan selama proses pengembangan. Metodologi XP dipilih karena pendekatannya yang fleksibel dan adaptif, dengan iterasi yang cepat dan kolaborasi yang intensif antara pengembang dan pengguna. Setiap fitur dikembangkan secara bertahap dengan melibatkan masukan langsung dari pengguna untuk memastikan bahwa sistem yang dihasilkan benar-benar sesuai dengan kebutuhan sekolah. Dengan pendekatan ini, proyek pengembangan modul Informasi Publik dan modul Evaluasi diharapkan dapat meningkatkan efisiensi, transparansi, dan kualitas pelayanan akademik di SMPN 29 Bandar Lampung, sekaligus memberikan pengalaman pengguna yang lebih baik melalui *platform* digital yang terintegrasi.

E. Nilai Bisnis

Adapun nilai bisnis yang dapat diperoleh dengan adanya Sistem ini diuraikan menjadi 2 poin, yaitu *tangible* dan *intangible*.

- *Tangible* (Terukur):
 - Peningkatan Efisiensi: Dengan sistem yang terintegrasi, proses pengelolaan data siswa, guru, dan pegawai menjadi lebih cepat dan mudah, sehingga mengurangi waktu yang dibutuhkan untuk pencatatan, pencarian, dan pengolahan

data. Serta proses evaluasi akademik, seperti penginputan nilai dan pembuatan *e-rapor*, menjadi lebih efisien karena otomatisasi dalam sistem.

- Pengurangan Biaya Operasional: Sistem ini mengurangi kebutuhan akan penggunaan dokumen fisik, seperti kertas dan buku catatan, yang secara langsung menghemat biaya operasional sekolah. Serta meminimalkan kesalahan pencatatan yang dapat menyebabkan pengeluaran tambahan untuk perbaikan data.
- Intangible (Tidak Terukur):
 - Peningkatan Kepuasan Pengguna: Siswa dan orang tua dapat dengan mudah mengakses informasi nilai, *e-rapor*, dan peringkat melalui sistem, sehingga memberikan kenyamanan dan transparansi dalam layanan akademik. Serta guru mendapatkan kemudahan dalam pengelolaan nilai dan data evaluasi, yang meningkatkan produktivitas kerja.
 - Transparansi Informasi: Modul Informasi Publik memastikan bahwa masyarakat dapat dengan mudah mendapatkan informasi terkait sekolah, seperti sejarah, prestasi, dan kegiatan sekolah. Hal ini meningkatkan kepercayaan masyarakat terhadap SMPN 29 Bandar Lampung.
 - Peningkatan Reputasi Sekolah: Adopsi teknologi berbasis *website* mencerminkan komitmen SMPN 29 Bandar Lampung dalam memberikan layanan pendidikan modern dan berkualitas, yang dapat meningkatkan reputasi sekolah sebagai institusi yang berorientasi pada teknologi.
 - Kemudahan Akses dan Keterjangkauan: Dengan sistem berbasis *website*, informasi dapat diakses kapan saja dan di

mana saja oleh pihak yang berkepentingan (guru, siswa, dan orang tua), menggunakan perangkat apa pun (komputer, tablet, atau *smartphone*).

F. Sponsor Proyek

Sponsor utama dalam pengembangan modul Informasi Publik dan modul Evaluasi pada sistem informasi akademik berbasis *website* untuk SMPN 29 Bandar Lampung adalah pihak internal sekolah, yaitu SMPN 29 Bandar Lampung sendiri, yang berlokasi di Jalan Soekarno-Hatta Bypass Sukarame, Bandar Lampung. Sebagai lembaga pendidikan, SMPN 29 Bandar Lampung memiliki komitmen untuk meningkatkan kualitas layanan akademik dan informasi publik melalui integrasi teknologi berbasis *web*. Kepala sekolah, Bapak Samsuri, S.Pd., bertindak sebagai pengarah utama dalam proyek ini, didukung oleh tim guru dan tenaga kependidikan yang terlibat dalam proses penyusunan kebutuhan sistem. Selain itu, sponsor proyek ini memberikan dukungan berupa sumber daya, informasi, dan pendanaan untuk memastikan kelancaran pengembangan sistem.

Tujuan utama sponsor dalam mendukung proyek ini adalah untuk menyediakan akses informasi yang transparan dan efisien, baik kepada siswa, guru, maupun masyarakat umum. Dengan pengembangan modul ini, SMPN 29 Bandar Lampung berharap dapat meningkatkan produktivitas dan efektivitas proses pengelolaan data akademik, sekaligus memberikan pengalaman pengguna yang lebih baik melalui sistem yang modern dan terintegrasi. Sebagai sponsor, SMPN 29 Bandar Lampung juga berperan aktif dalam memberikan masukan selama proses pengembangan sistem menggunakan metode *Extreme Programming* (XP), sehingga setiap fitur yang dikembangkan benar-benar sesuai dengan kebutuhan sekolah.

G. Kebutuhan Bisnis

Berdasarkan kebutuhan untuk meningkatkan efisiensi dan kualitas layanan dalam pengelolaan data akademik di SMPN 29 Bandar Lampung, diperlukan sistem informasi berbasis *web* yang dapat mendukung pengelolaan data akademik secara lebih terintegrasi, cepat, dan andal. Sistem ini bertujuan untuk memudahkan akses informasi bagi siswa, guru, dan orang tua, serta mendukung proses evaluasi akademik yang lebih transparan dan efisien. Modul Informasi Publik diperlukan untuk menyediakan berbagai informasi penting, seperti sejarah sekolah, visi dan misi, struktur organisasi, ekstrakurikuler, prestasi, dan galeri kegiatan. Modul ini berfungsi sebagai sarana komunikasi yang efektif antara sekolah dan masyarakat umum, sehingga menciptakan transparansi dalam penyampaian informasi sekolah. Sementara itu, Modul Evaluasi dirancang untuk mempermudah proses pengelolaan data nilai siswa, pembuatan *e-rapor*, serta pemantauan peringkat siswa. Modul ini diharapkan dapat mempercepat proses input data oleh guru dan memberikan akses yang mudah bagi siswa dan orang tua untuk memantau perkembangan akademik secara *real-time*. Dengan adanya sistem informasi ini, diharapkan pengelolaan data akademik di SMPN 29 Bandar Lampung menjadi lebih terstruktur dan mendukung efisiensi dalam pelayanan pendidikan, baik bagi internal sekolah maupun masyarakat umum.

H. Persyaratan Bisnis

Kebutuhan bisnis pada pengembangan sistem informasi ini terdiri dari kebutuhan non fungsional dan kebutuhan fungsional. Kebutuhan non fungsional yaitu:

1. Sistem harus dirancang agar mudah digunakan oleh semua pengguna, termasuk guru, siswa, dan orang tua tanpa memerlukan pelatihan khusus
2. Sistem harus kompatibel dengan *browser* populer seperti Google Chrome, Mozilla Firefox, dan Microsoft Edge
3. Sistem harus mampu berjalan dengan baik secara *online*, melalui *website*
4. Antarmuka sistem harus responsif. Sistem harus dapat diakses dari berbagai perangkat, termasuk komputer, tablet, dan ponsel, melalui *browser* modern
5. Sistem harus mampu memuat halaman dalam waktu kurang dari 10 detik untuk memastikan pengalaman pengguna yang baik
6. Sistem harus menerapkan mekanisme *login* berbasis *username* dan *password* yang terenkripsi untuk memastikan hanya pengguna yang berwenang yang dapat mengakses modul tertentu

Sedangkan kebutuhan fungsional adalah sebagai berikut:

1. User dapat mengakses nilai, jadwal mata pelajaran, data siswa, data guru dan pegawai, prestasi, riwayat administrasi, galeri kegiatan, ekstrakurikuler, sejarah singkat, hymne, visi misi, struktur organisasi, *ranking*, *e-raport*, dan kemajuan belajar
2. Admin dapat mengelola data siswa, data guru dan pegawai, nilai, jadwal mata pelajaran, riwayat administrasi, galeri kegiatan, prestasi, sejarah singkat, dan ekstrakurikuler
3. Guru dapat mengelola nilai siswa

I. Masalah Khusus / Kendala

Masalah khusus / kendala yang dihadapi dalam pengembangan sistem informasi ini adalah:

- Kompatibilitas: Sistem harus dapat dijalankan di berbagai perangkat seperti komputer, *smartphone* dan tablet.
- Keamanan: Memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses fitur-fitur tertentu sesuai dengan peran mereka.
- Estetika: Sistem harus memiliki tampilan yang nyaman dilihat dan ramah pengguna untuk memastikan sistem dapat dimengerti semua kalangan

J. Persyaratan Non Fungsional

1. Persyaratan Operasional:

- Sistem harus dapat diakses secara *online* baik melalui *website*.
- Sistem harus dapat dijalankan di perangkat *smartphone*, tablet, maupun komputer.

2. Persyaratan Kinerja:

- Sistem harus mampu memproses penggunaan fitur, seperti akses data akademik, dan unggahan nilai pembelajaran, dalam waktu tidak lebih dari 30 detik per permintaan.

3. Persyaratan Keandalan:

- Sistem harus memiliki *uptime* 99.9% untuk memastikan ketersediaan layanan yang konsisten bagi pengguna.

4. Persyaratan Keselamatan:

- Enkripsi data harus diterapkan untuk semua data sensitif seperti *password* akun pengguna

5. Persyaratan Keamanan:

- memastikan bahwa setiap fitur hanya dapat diakses oleh pengguna yang memiliki peran dan otorisasi sesuai dengan hak akses yang telah ditentukan.

6. Persyaratan Kegunaan:

- Antarmuka pengguna harus intuitif dan mudah digunakan, dengan desain yang responsif untuk berbagai perangkat.

K. Persyaratan Fungsional

Sistem informasi yang dibangun memiliki kebutuhan fungsional sebagai berikut:

- **Penilaian:** Guru dapat menambah, menghapus, dan mengubah nilai siswa.
- **Manajemen Data Akademik dan Informasi Publik:** Admin dapat menambah, mengubah, dan menghapus data siswa, data guru dan pegawai, jadwal mata pelajaran, riwayat administrasi, sejarah singkat, ekstrakurikuler, galeri kegiatan, dan prestasi
- **Pembelajaran:** Pengguna dapat mengakses nilai, kemajuan belajar, data siswa, data guru dan pegawai, jadwal mata pelajaran, riwayat administrasi, informasi publik, *ranking*, dan *e-raport*

L. Hasil Kegiatan Wawancara

Berdasarkan hasil wawancara, ditemukan bahwa pengelolaan informasi publik di SMPN 29 Bandar Lampung masih menghadapi beberapa kendala, seperti proses penyampaian informasi publik yang dilakukan secara manual atau melalui media sosial yang tidak terintegrasi, sehingga masyarakat umum dan orang tua siswa sering kesulitan mendapatkan informasi terbaru. Selain itu, tidak adanya sistem terpusat menyebabkan

sekolah kesulitan dalam mengelola informasi seperti menambah, mengedit, atau menghapus data secara cepat dan efisien.

Di sisi lain, pengelolaan evaluasi akademik juga mengalami hambatan. Guru menghadapi kesulitan dalam mencatat nilai harian, melakukan rekapitulasi nilai, dan menyusun rapor secara manual. Proses penghitungan peringkat siswa yang dilakukan tanpa sistem otomatisasi rentan terhadap kesalahan dan memakan waktu. Orang tua dan siswa juga tidak memiliki akses langsung untuk memantau hasil belajar secara *real-time*, sehingga informasi hanya tersedia pada saat penerimaan rapor.

Guru dan staf administrasi mengungkapkan kebutuhan akan sistem yang dapat mempermudah pengelolaan data akademik, seperti input nilai siswa, pembuatan *e-rapor* secara otomatis, dan pengelolaan data siswa, guru, serta pegawai. Mereka menginginkan sistem yang dapat mendukung pembaruan, dan pencarian data dengan lebih mudah.

Selain itu, siswa dan orang tua mengharapkan adanya sistem yang transparan dan mudah diakses untuk melihat informasi akademik seperti nilai, peringkat, dan pemberitahuan penting lainnya.

Hasil wawancara juga mengungkapkan kebutuhan tambahan, yaitu perlunya antarmuka pengguna yang intuitif dan mudah digunakan oleh semua pihak, termasuk guru, siswa, orang tua, dan staf administrasi. Sistem ini juga harus mendukung aksesibilitas di berbagai perangkat, seperti komputer, tablet, dan *smartphone*, sehingga pengguna dapat mengaksesnya kapan saja dan di mana saja.

M. Hasil Kegiatan Observasi

Berdasarkan hasil observasi yang dilakukan terhadap pengelolaan informasi publik dan evaluasi akademik di SMPN 29 Bandar Lampung, ditemukan beberapa masalah signifikan baik dari sisi guru, staf administrasi, siswa, maupun orang tua, yaitu:

1. Pengelolaan Informasi Publik:

- Guru dan Staf Administrasi:

Sulit dalam menyampaikan informasi publik secara terpusat, seperti sejarah sekolah, visi dan misi, kegiatan ekstrakurikuler, dan prestasi. Saat ini, informasi disampaikan secara manual atau melalui media sosial, yang sering kali tidak konsisten dan tidak terintegrasi.

- Orang Tua dan Siswa:

Kesulitan mendapatkan informasi terbaru tentang sekolah karena pembaruan informasi tidak dilakukan secara terstruktur dan tidak tersedia dalam satu *platform* yang mudah diakses.

2. Pengelolaan Evaluasi Akademik:

- Guru:

Mengalami kesulitan dalam mencatat nilai harian, UTS, dan UAS, serta proses rekapitulasi nilai yang masih dilakukan secara manual, yang rentan terhadap kesalahan.

- Orang Tua dan Siswa:

Tidak memiliki akses langsung untuk memantau hasil belajar siswa, sehingga informasi nilai atau peringkat siswa hanya dapat diketahui setelah pembagian rapor.

3. Pengelolaan Data Guru, Pegawai, dan Siswa:

- Staf Administrasi:

Menghadapi tantangan dalam mengelola data guru, pegawai, dan siswa, termasuk pendaftaran, pembaruan,

dan pencarian data. Proses pengelolaan data dilakukan secara manual, yang memakan waktu dan rentan terhadap duplikasi atau kehilangan data.

2. Design (Perancangan)

A. CRC Cards

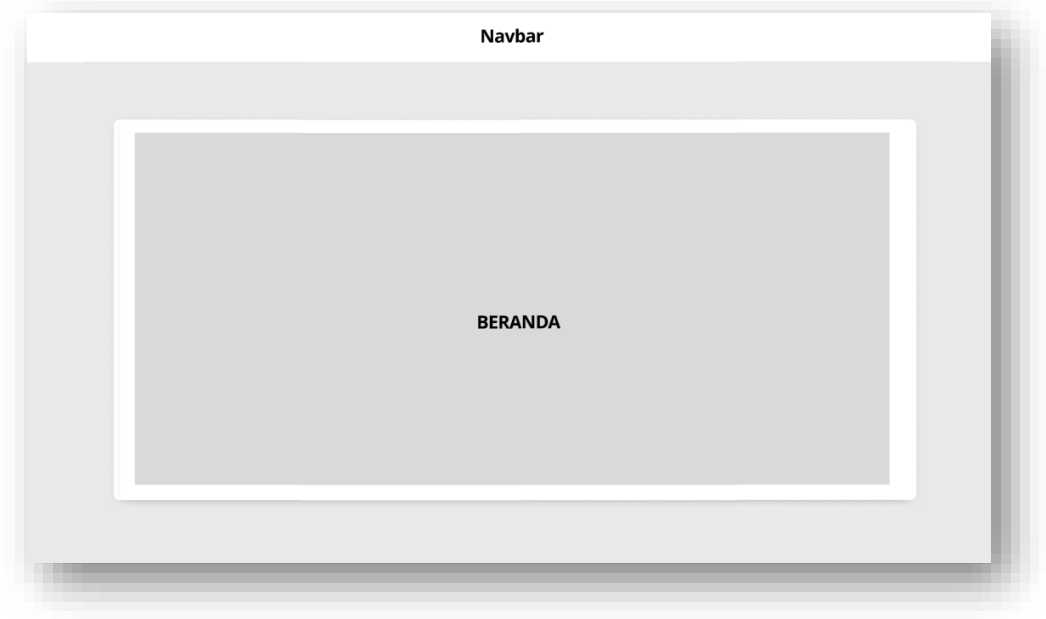
USERS ✕ • Menyimpan data login pengguna sistem (email, username, password)	AUTH_GROUPS_USERS ✕ • AUTH_GROUPS_USERS
AUTH_GROUPS ✕ • Menyimpan daftar grup/rol pengguna sistem	AUTH_GROUPS_USERS ✕ • AUTH_GROUPS_USERS
AUTH_GROUPS_USERS ✕ • Menyimpan relasi many-to-many antara grup dan user	USERS ✕ • USERS
JADWAL ✕ • Menyimpan informasi jadwal pelajaran berdasarkan hari dan kelas	PEGAWAI ✕ • Menyimpan data pegawai sekolah termasuk jabatan dan status
GURU ✕ • Menyimpan data guru dan informasi kontak	
SEJARAH ✕ • Menyimpan catatan sejarah kepemimpinan	PRESTASI ✕ • Menyimpan data prestasi yang diraih oleh siswa atau sekolah
EKSTRAKURIKULER ✕ • Menyimpan data kegiatan ekstrakurikuler	
GALERIKEGIATAN ✕ • Menyimpan dokumentasi kegiatan sekolah	SISWA ✕ • Menyimpan data pribadi siswa • Menyimpan info kelas & semester saat ini
RIWAYAT_ADMINISTRASI ✕ • Menyimpan catatan pembayaran siswa • Mencatat tanggal dan nominal pembayaran	SISWA ✕ • RIWAYAT_ADMINISTRASI • NILAI • RIWAYAT_NILAI • NILAI_AKHIR • RIWAYAT_NILAI_AKHIR
NILAI ✕ • Menyimpan nilai siswa dari tiap penilaian (PH1-UAS) • Menghitung & menyimpan rata-rata nilai	SISWA ✕ • SISWA
RIWAYAT_NILAI ✕ • Menyimpan nilai historis siswa dari semester sebelumnya	SISWA ✕ • SISWA • NILAI
NILAI_AKHIR ✕ • Menyimpan nilai akhir siswa per semester & kelas	SISWA ✕ • SISWA • NILAI
RIWAYAT_NILAI_AKHIR ✕ • Menyimpan data nilai akhir historis dari semester sebelumnya	SISWA ✕ • SISWA • NILAI

Gambar 4. CRC Cards

CRC Cards ini menjelaskan struktur kelas dalam sistem informasi akademik dan tugas utama tiap kelas serta bagaimana mereka bekerja sama. Kelas users menyimpan data login seperti email, username, dan password, dan berhubungan dengan kelas auth_groups_users untuk mengatur peran pengguna. Karena setiap pengguna hanya punya satu peran, hubungan antara users dan auth_groups_users bersifat one to one. Kelas auth_group menyimpan daftar peran yang bisa dimiliki pengguna. Kelas jadwal mengatur jadwal pelajaran berdasarkan hari dan kelas. Data pegawai dan guru disimpan di kelas pegawai dan guru, yang mencakup informasi jabatan dan kontak. Kelas sejarah, prestasi, ekstrakurikuler, dan galerikegiatan bertanggung jawab menyimpan informasi sejarah sekolah, prestasi, kegiatan ekstrakurikuler, dan foto kegiatan sekolah. Kelas siswa menyimpan data pribadi siswa dan informasi kelas saat ini, serta bekerja sama dengan kelas riwayat_administrasi yang mencatat pembayaran siswa, dan kelas seperti nilai, riwayat_nilai, nilai_akhir, dan riwayat_nilai_akhir menyimpan nilai dan catatan nilai siswa dari semester sebelumnya. Secara keseluruhan, CRC Cards ini bertujuan membantu memahami fungsi tiap kelas dan bagaimana mereka saling terhubung untuk membuat sistem akademik yang teratur dan efektif.

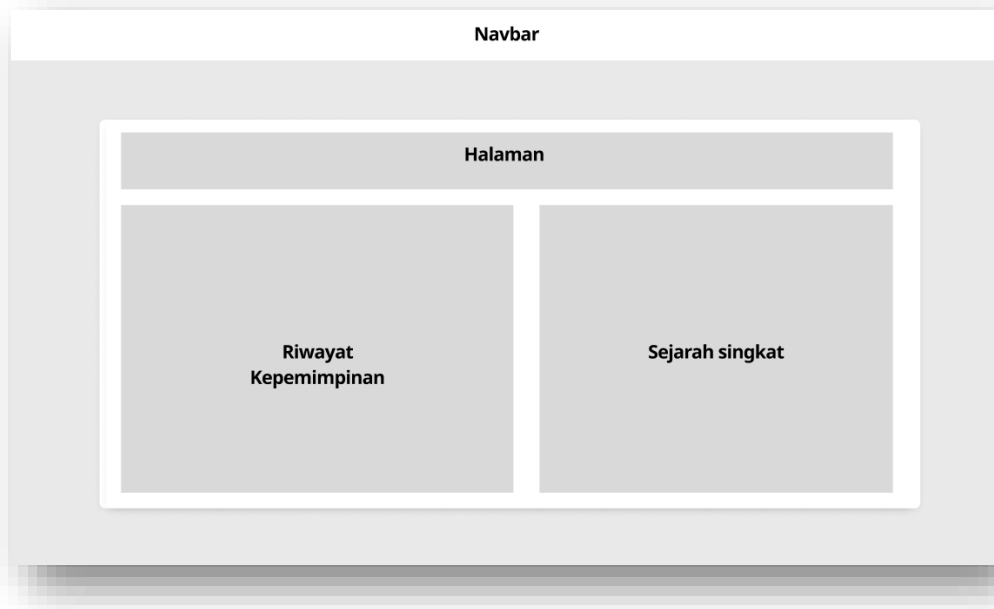
B. Prototype

1. Halaman Beranda



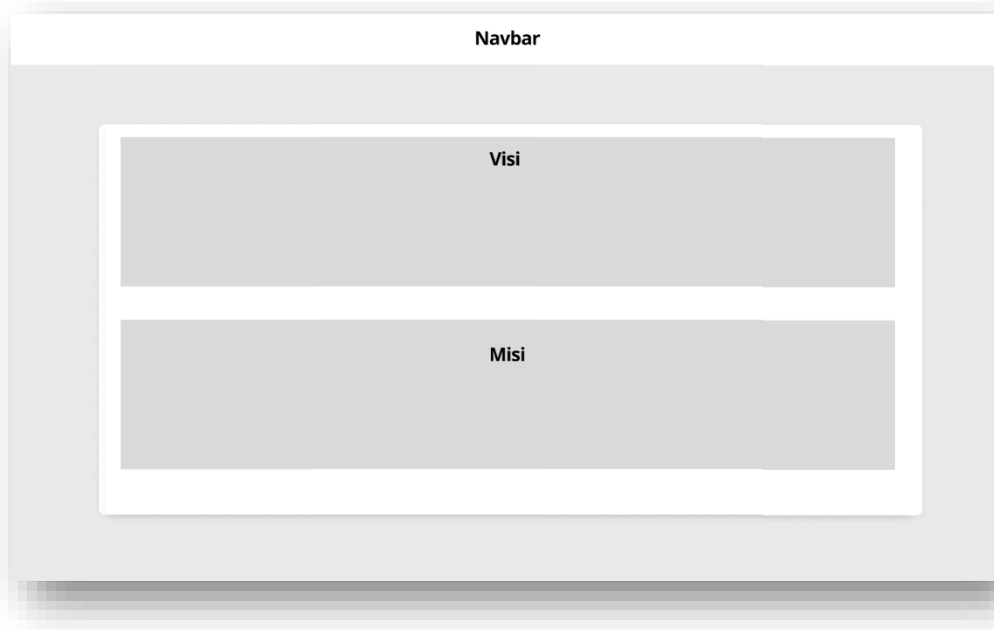
Gambar 5. Halaman Beranda

2. Halaman Sejarah Singkat



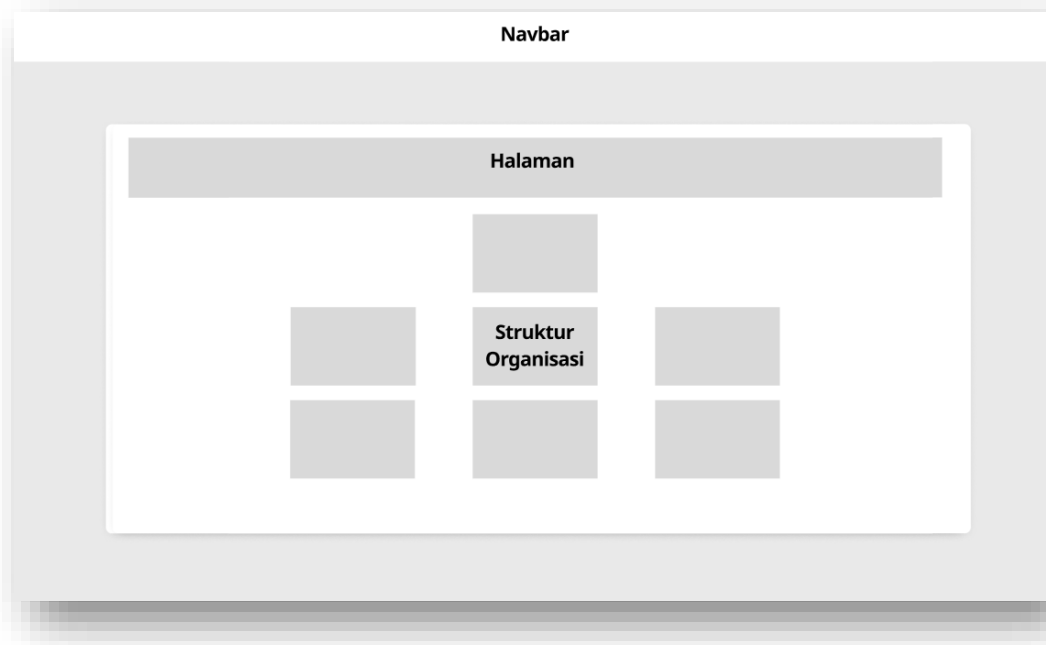
Gambar 6. Halaman Sejarah Singkat

3. Halaman Visi Misi



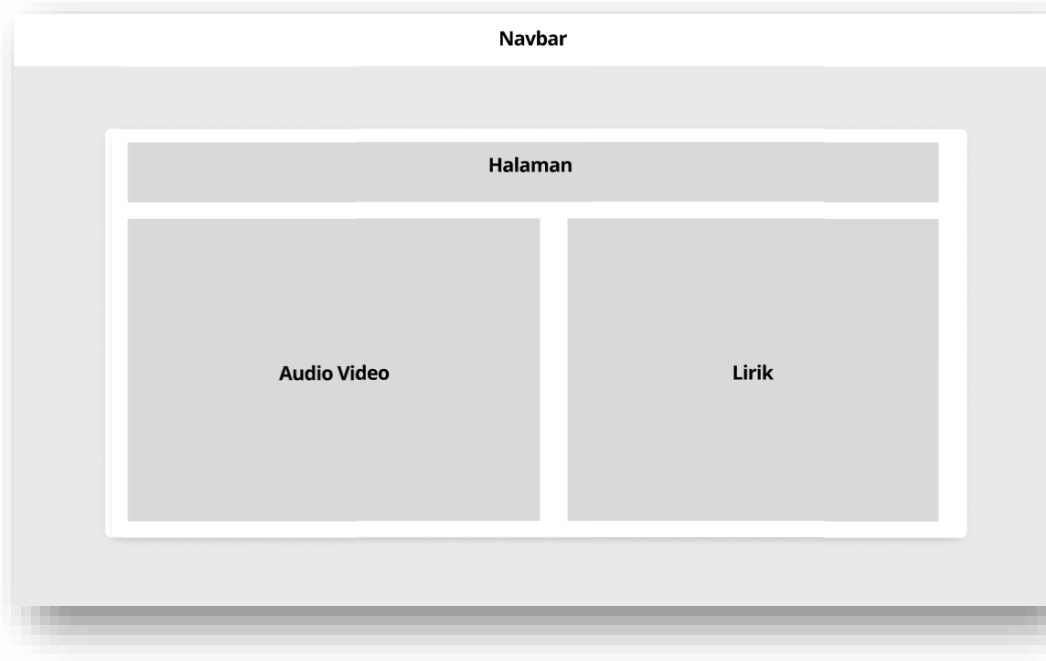
Gambar 7. Halaman Visi Misi

4. Halaman Struktur Organisasi



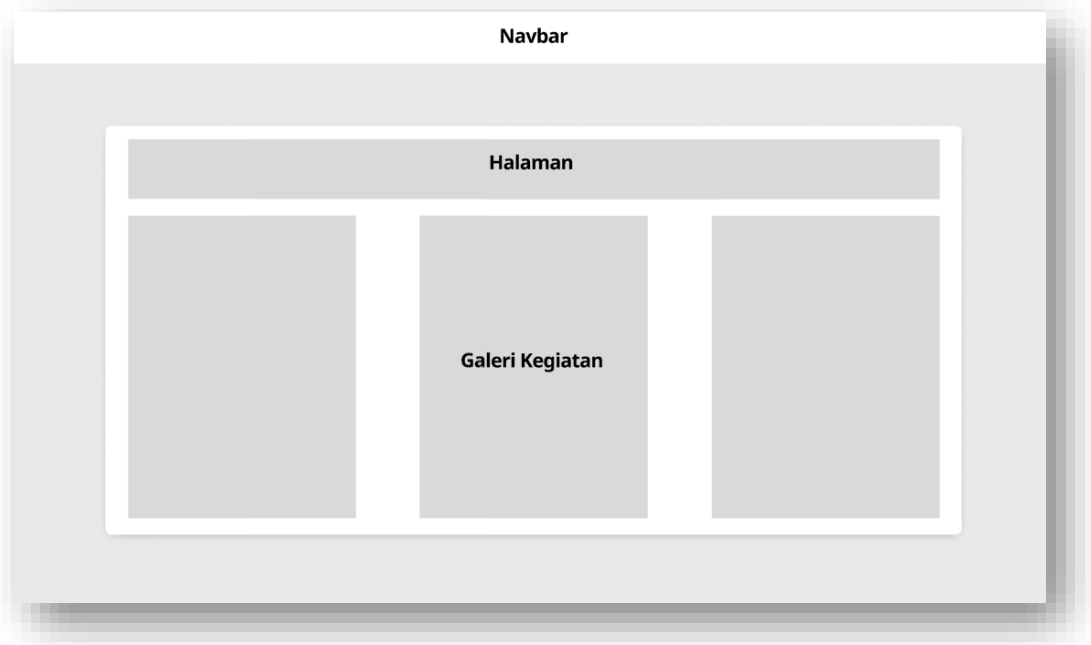
Gambar 8. Halaman Struktur Organisasi

5. Halaman Hymne



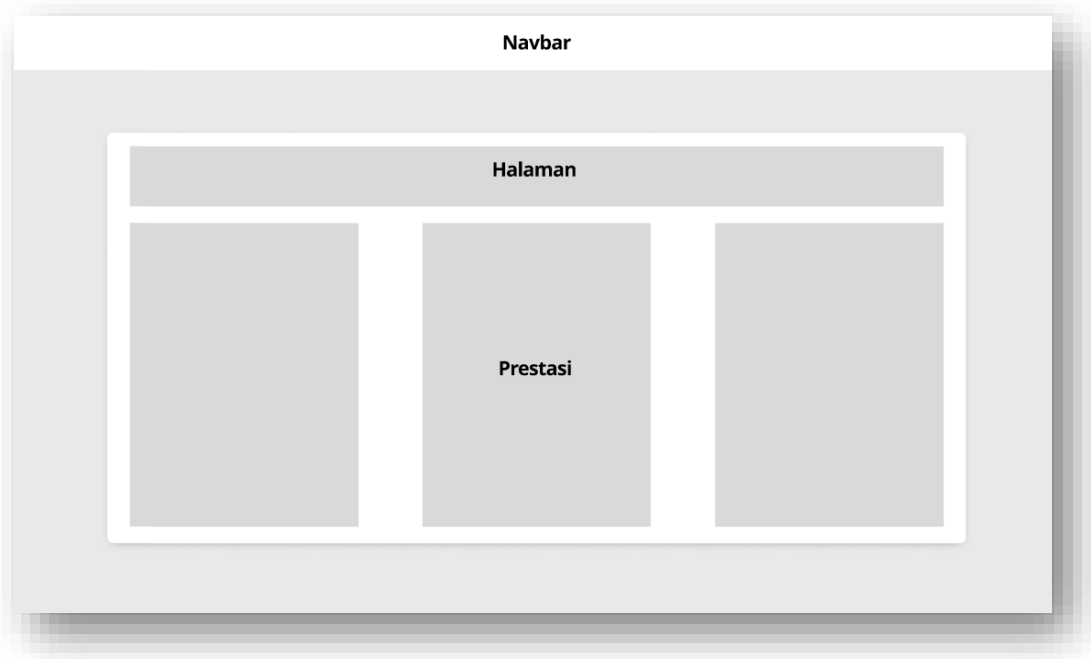
Gambar 9. Halaman Hymne

6. Halaman Galeri Kegiatan



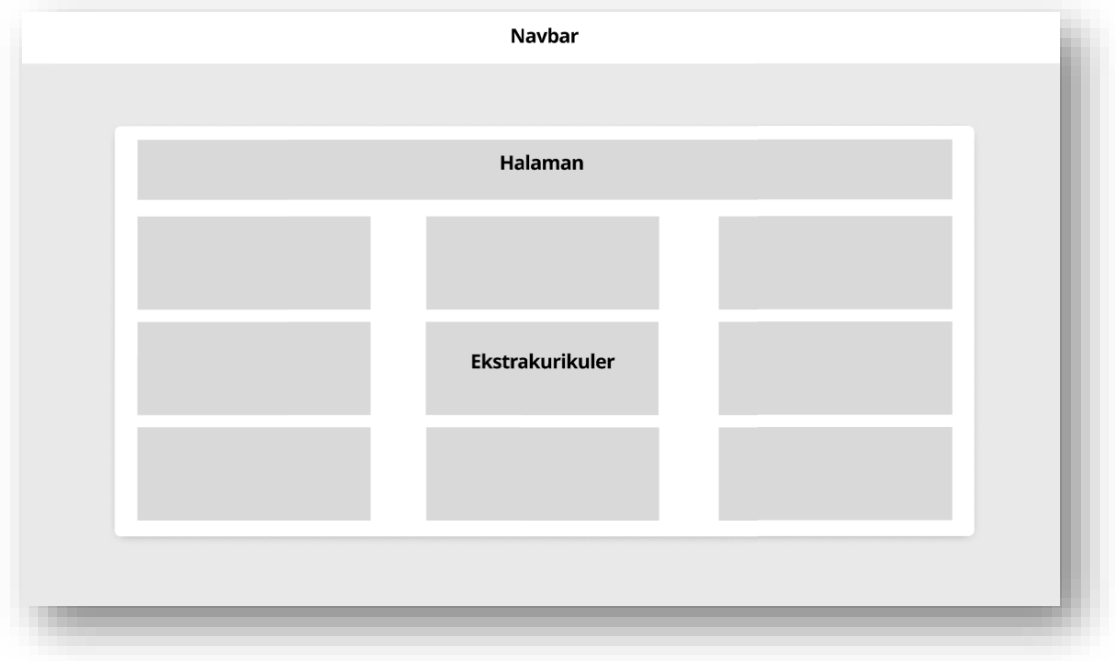
Gambar 10. Halaman Galeri Kegiatan

7. Halaman Prestasi



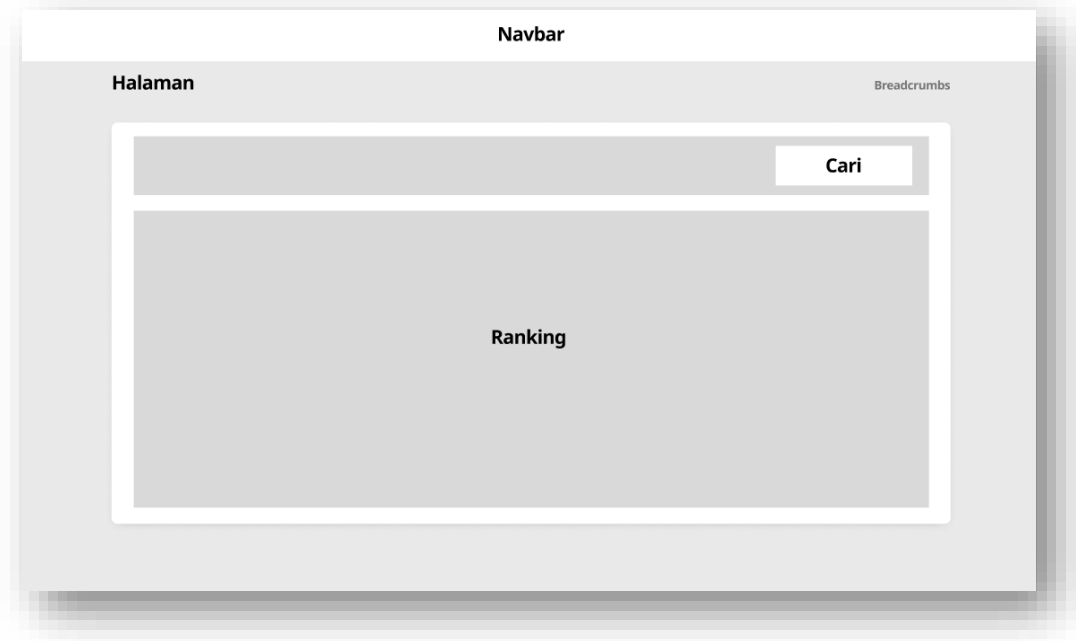
Gambar 11. Halaman Prestasi

8. Halaman Ekstrakurikuler



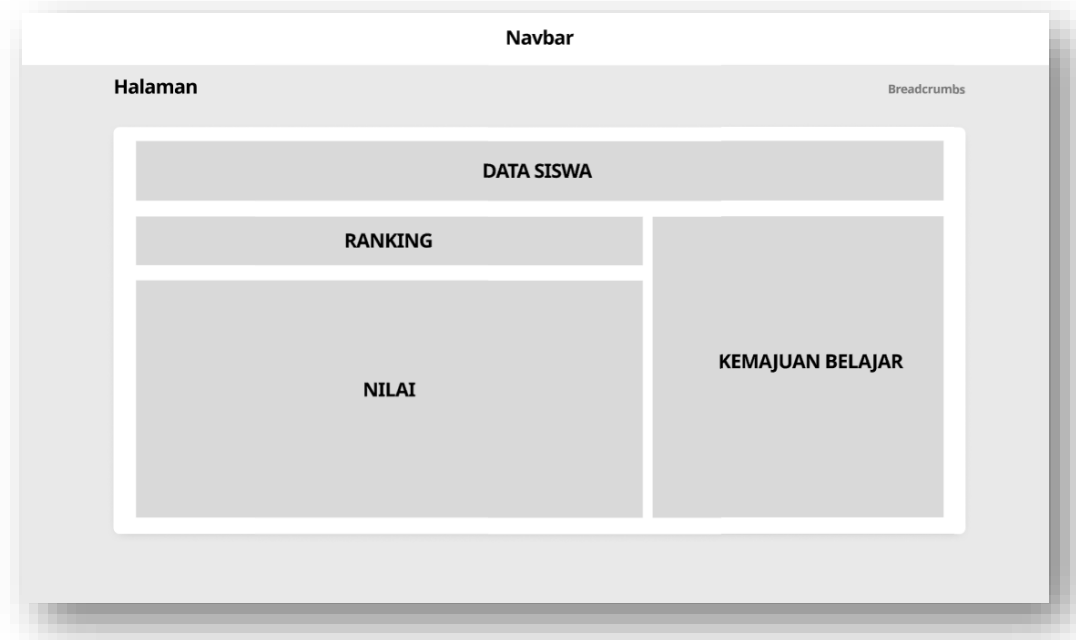
Gambar 12. Halaman Ekstrakurikuler

9. Halaman *Ranking*



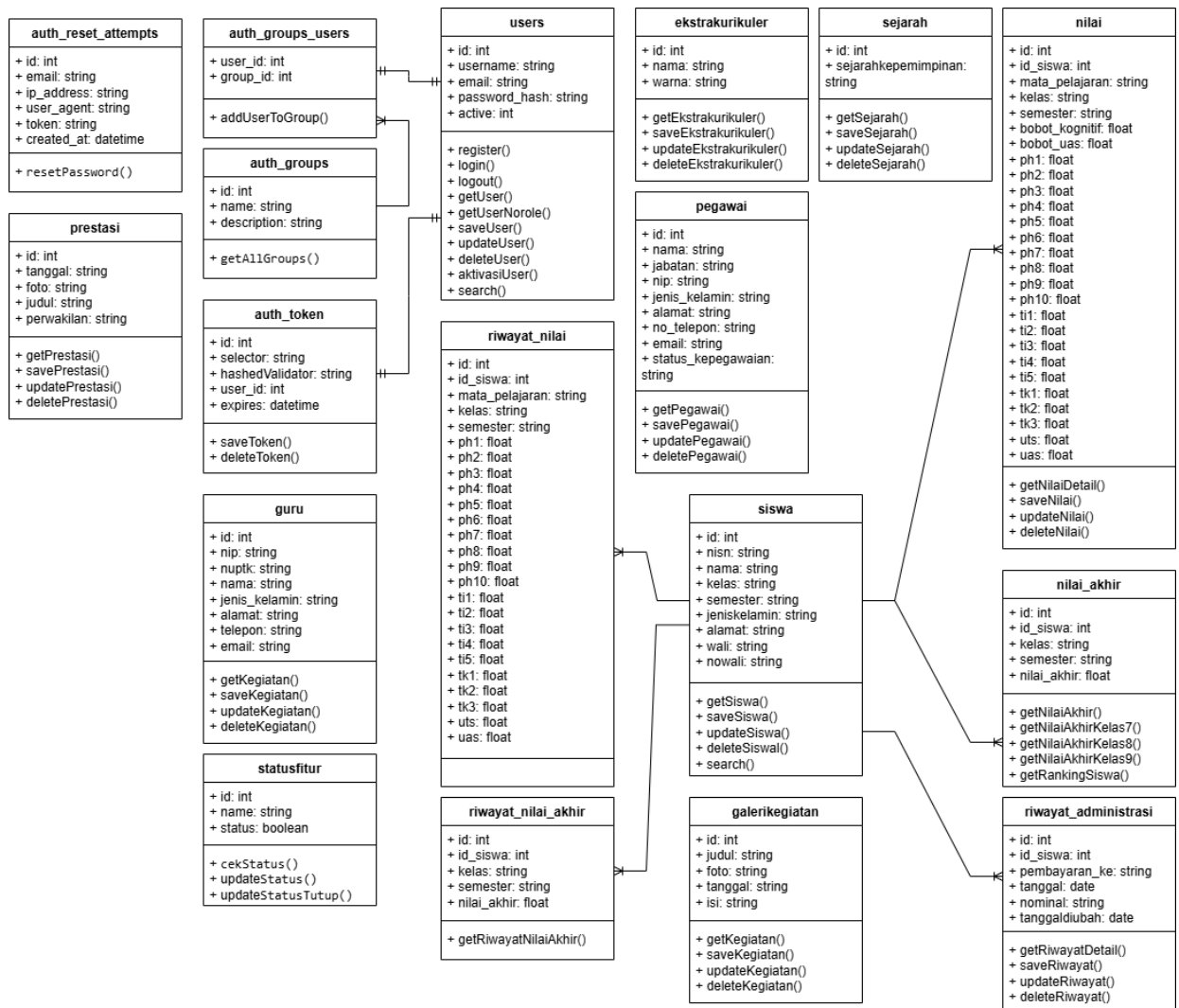
Gambar 13. Halaman *Ranking*

10. Halaman *E-Raport*



Gambar 14. Halaman *E-Raport*

C. Class Diagram



Gambar 15. Class Diagram

Class diagram ini menggambarkan struktur data dan hubungan antar komponen utama dalam sistem. Kelas *users* menyimpan data pengguna sistem seperti *username*, *email*, dan *password*, serta menyediakan fungsi-fungsi untuk login, pengelolaan data pengguna, dan pencarian. Kelas *auth_groups* dan *auth_groups_users* mengelola peran pengguna dalam sistem dengan menghubungkan pengguna ke grup tertentu. Kelas *auth_token* menyimpan informasi token otentikasi pengguna,

termasuk ID, selector, hashed validator, dan tanggal kedaluwarsa, dengan fungsi untuk menyimpan dan menghapus token. Kelas `auth_reset_attempts` mencatat upaya reset kata sandi pengguna, mencakup ID pengguna, email, alamat IP, token, dan waktu pembuatan reset, serta menyediakan fungsi untuk mengatur reset password. Kedua kelas ini mendukung keamanan dan pemulihan akses pengguna.

Kelas `siswa` memuat data lengkap siswa, mulai dari identitas seperti NISN, nama, jenis kelamin, alamat, hingga informasi wali dan status kelas serta semester. Kelas ini juga terhubung dengan beberapa kelas lainnya. Yakni, terhubung dengan kelas `nilai` yang menyimpan nilai rinci per mata pelajaran dan berbagai jenis penilaian. Terhubung dengan kelas `riwayat_nilai` yang mencatat nilai historis siswa. Terhubung dengan kelas `nilai_akhir` yang menyimpan nilai akhir siswa setiap semester. Dan terhubung dengan kelas `riwayat_nilai_akhir` sebagai arsip nilai akhir dari semester sebelumnya. Kelas-kelas nilai ini saling terkait dengan kelas `siswa` untuk memastikan data nilai terintegrasi dan akurat. Serta kelas `siswa` ini juga terhubung dengan kelas `riwayat_administrasi` yang mengelola data pembayaran siswa, termasuk tanggal, jumlah, dan riwayat transaksi administrasi.

Sedangkan ada pula kelas yang berdiri sendiri, tidak berelasi dengan kelas lain. Diantaranya, Kelas `pegawai` dan `guru` yang menyimpan data staf sekolah, seperti jabatan, NIP, jenis kelamin, kontak, dan status kepegawaian, serta menyediakan metode untuk mengelola data tersebut secara lengkap. Kelas `prestasi` mencatat prestasi yang diraih oleh siswa atau sekolah, lengkap dengan tanggal, foto, dan detail prestasi. Kelas `ekstrakurikuler` mengelola data ketersediaan ekstrakurikuler. Kelas `galerikegiatan` menyimpan dokumentasi kegiatan sekolah dalam bentuk foto dan

deskripsi. Dan terakhir kelas sejarah bertugas menyimpan catatan sejarah kepemimpinan sekolah.

Semua kelas ini saling berhubungan dengan baik untuk memungkinkan pengambilan, penyimpanan, pembaruan, dan penghapusan data secara efisien.

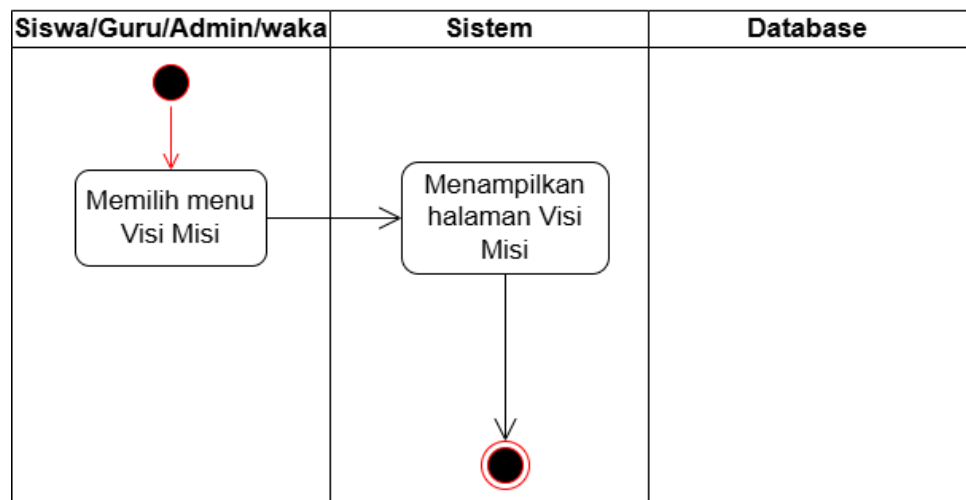
D. *Activity Diagram*

Selanjutnya dibuat *Activity Diagram* berdasarkan *Use Case* yang telah dibuat sebelumnya. Adapun *Activity Diagram* adalah sebagai berikut

1. *Activity Diagram* Melihat Visi Misi

Activity diagram yang ditunjukkan pada Gambar 16 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat visi dan misi pada sistem. Proses diawali dengan pengguna memilih menu "Visi Misi" yang tersedia di dalam sistem. Setelah menu dipilih, sistem akan menampilkan halaman yang berisi informasi mengenai visi dan misi. Setelah halaman tersebut ditampilkan, proses pun selesai. Diagram ini menunjukkan alur sederhana tanpa adanya interaksi lebih lanjut dengan *database*, karena informasi visi dan misi hanya ditampilkan tanpa adanya manipulasi data.

Activity Diagram Melihat Visi Misi

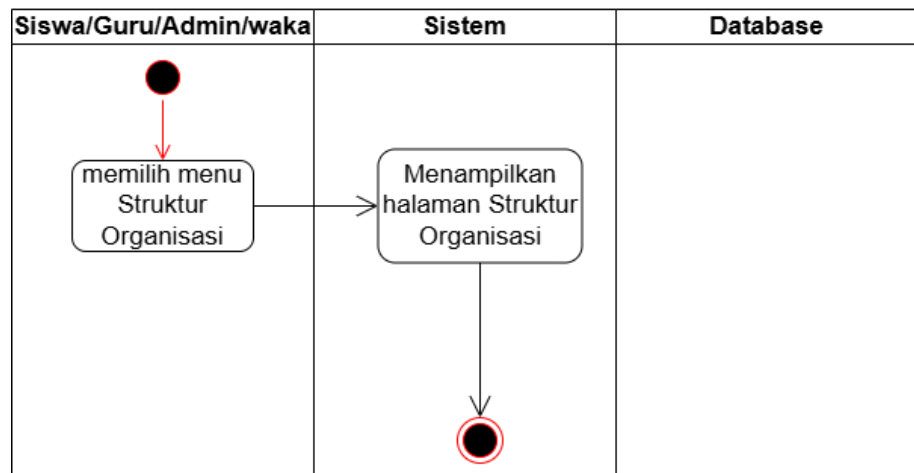


Gambar 16. *Activity Diagram* Melihat Visi Misi

2. *Activity Diagram* Melihat Struktur Organisasi

Activity diagram yang ditunjukkan pada Gambar 17 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat struktur organisasi di dalam sistem. Proses dimulai ketika siswa memilih menu "Struktur Organisasi" yang tersedia dalam sistem. Setelah itu, sistem akan menampilkan halaman yang berisi informasi mengenai struktur organisasi. Setelah halaman tersebut ditampilkan, proses selesai tanpa adanya interaksi lebih lanjut dengan *database*. Diagram ini menunjukkan bahwa proses melihat struktur organisasi berlangsung secara langsung dan sederhana, hanya melibatkan interaksi antara pengguna dan sistem tanpa perlu validasi atau manipulasi data.

Activity Diagram Melihat Struktur Organisasi

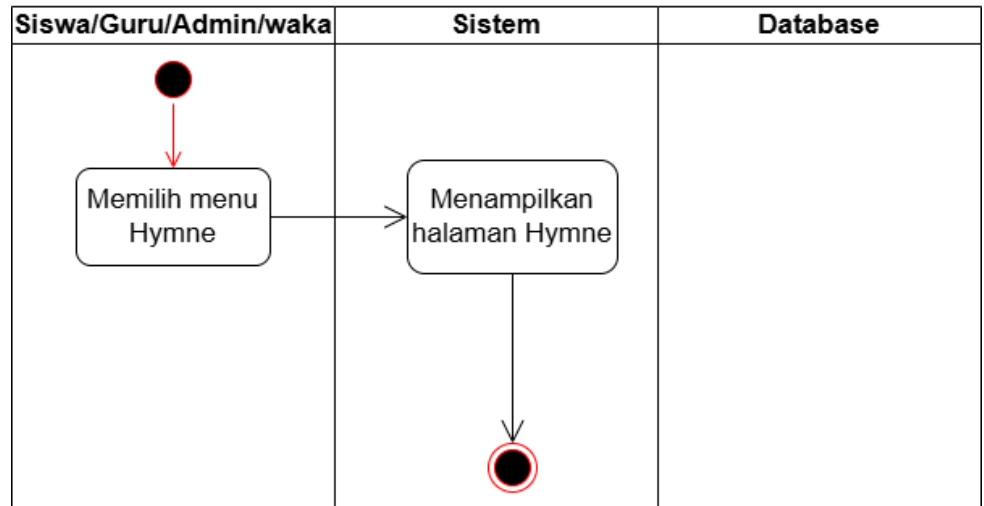


Gambar 17. Activity Diagram Melihat Struktur Organisasi

3. Activity Diagram Melihat Hymne

Activity diagram yang ditunjukkan pada Gambar 18 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat halaman Hymne di dalam sistem. Proses dimulai ketika pengguna memilih menu "Hymne" yang tersedia di sistem. Setelah itu, sistem akan menampilkan halaman Hymne yang berisi informasi atau lirik lagu Hymne yang diinginkan. Setelah halaman tersebut ditampilkan, proses selesai tanpa adanya interaksi lebih lanjut dengan *database*. Diagram ini menunjukkan bahwa proses melihat Hymne berjalan dengan sederhana, hanya melibatkan pemilihan menu dan tampilan informasi oleh sistem.

Activity Diagram Melihat Hymne

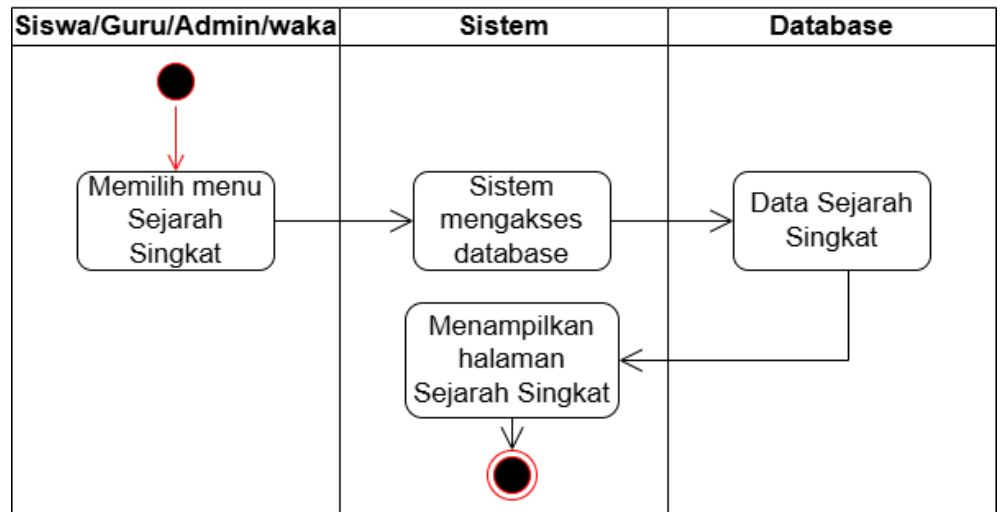


Gambar 18. *Activity Diagram* Melihat Hymne

4. *Activity Diagram* Melihat Sejarah Singkat

Activity diagram yang ditunjukkan pada Gambar 19 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat informasi sejarah singkat di dalam sistem. Proses dimulai ketika pengguna memilih menu "Sejarah Singkat" yang tersedia di sistem. Setelah itu, sistem akan mengakses *database* untuk mengambil data sejarah singkat yang telah tersimpan. Setelah data berhasil diperoleh, sistem menampilkan halaman sejarah singkat yang berisi informasi tersebut. Proses ini memastikan bahwa informasi yang ditampilkan selalu berasal dari data yang tersimpan di *database*, sehingga pengguna mendapatkan informasi yang akurat dan terkini.

Activity Diagram Melihat Sejarah Singkat

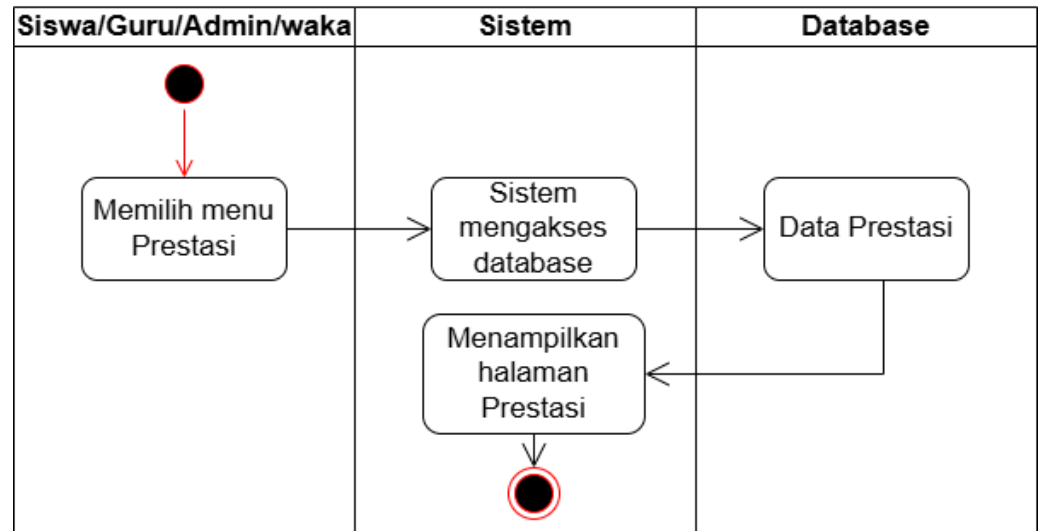


Gambar 19. *Activity Diagram* Melihat Sejarah Singkat

5. *Activity Diagram* Melihat Prestasi

Activity diagram yang ditunjukkan pada Gambar 20 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat informasi prestasi di dalam sistem. Proses dimulai ketika pengguna memilih menu "Prestasi" yang tersedia di sistem. Setelah itu, sistem akan mengakses *database* untuk mengambil data prestasi yang telah tersimpan. Setelah data berhasil diperoleh, sistem menampilkan halaman prestasi yang berisi informasi mengenai pencapaian yang telah dicatat. Proses ini memastikan bahwa informasi yang ditampilkan selalu diperbarui dan diambil langsung dari *database*, sehingga pengguna dapat melihat daftar prestasi dengan akurat dan terpercaya.

Activity Diagram Melihat Prestasi

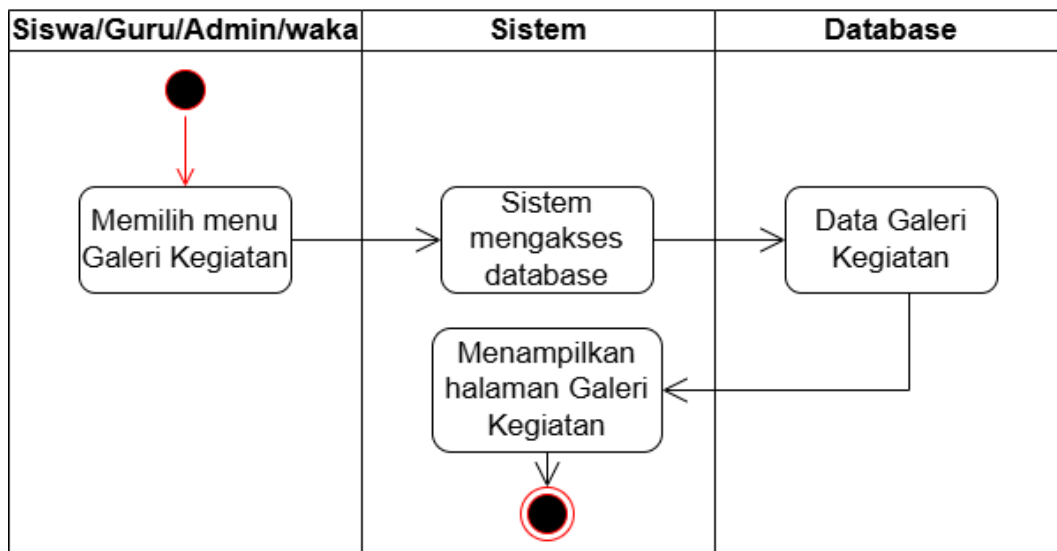


Gambar 20. *Activity Diagram* Melihat Prestasi

6. *Activity Diagram* Melihat Galeri Kegiatan

Activity diagram yang ditunjukkan pada Gambar 21 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat galeri kegiatan di dalam sistem. Proses diawali ketika pengguna memilih menu "Galeri Kegiatan" yang tersedia di sistem. Setelah itu, sistem akan mengakses *database* untuk mengambil data galeri kegiatan yang telah tersimpan. Setelah data berhasil diperoleh, sistem akan menampilkan halaman galeri kegiatan yang berisi berbagai dokumentasi kegiatan yang telah dilakukan. Proses ini memastikan bahwa pengguna dapat melihat informasi dan gambar dari berbagai kegiatan yang telah diunggah ke dalam sistem dengan akurat dan mudah diakses.

Activity Diagram Melihat Galeri Kegiatan

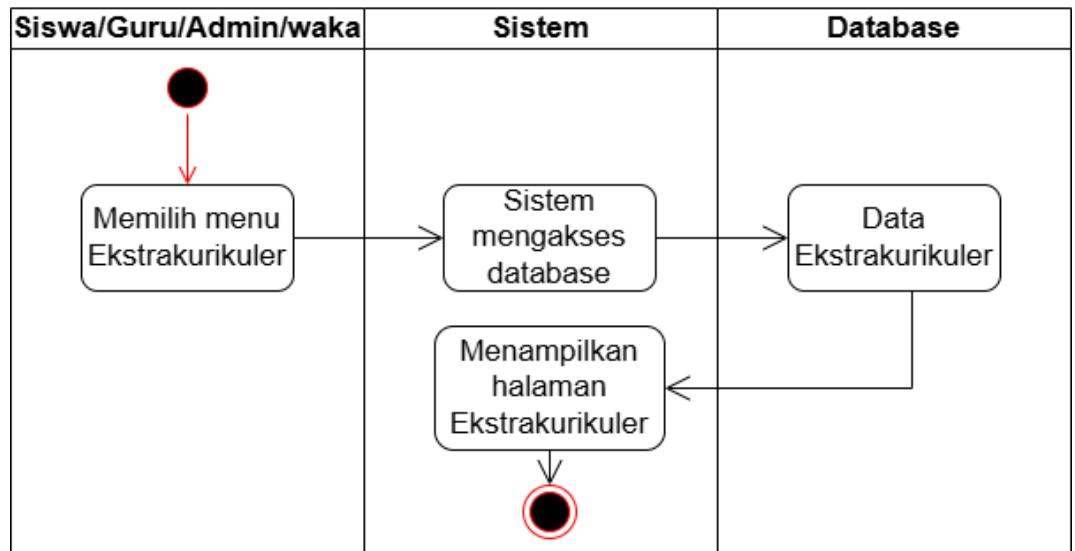


Gambar 21. *Activity Diagram* Melihat Galeri Kegiatan

7. *Activity Diagram* Melihat Ekstrakurikuler

Activity diagram yang ditunjukkan pada Gambar 22 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat informasi ekstrakurikuler di dalam sistem. Proses dimulai ketika pengguna memilih menu "Ekstrakurikuler" yang tersedia di sistem. Setelah itu, sistem akan mengakses *database* untuk mengambil data terkait ekstrakurikuler yang telah tersimpan. Setelah data berhasil diperoleh, sistem akan menampilkan halaman ekstrakurikuler yang berisi informasi mengenai berbagai kegiatan ekstrakurikuler yang tersedia. Proses ini memastikan bahwa pengguna dapat dengan mudah mengakses informasi mengenai pilihan ekstrakurikuler yang ada di sekolah melalui sistem secara akurat dan terkini.

Activity Diagram Melihat Ekstrakurikuler

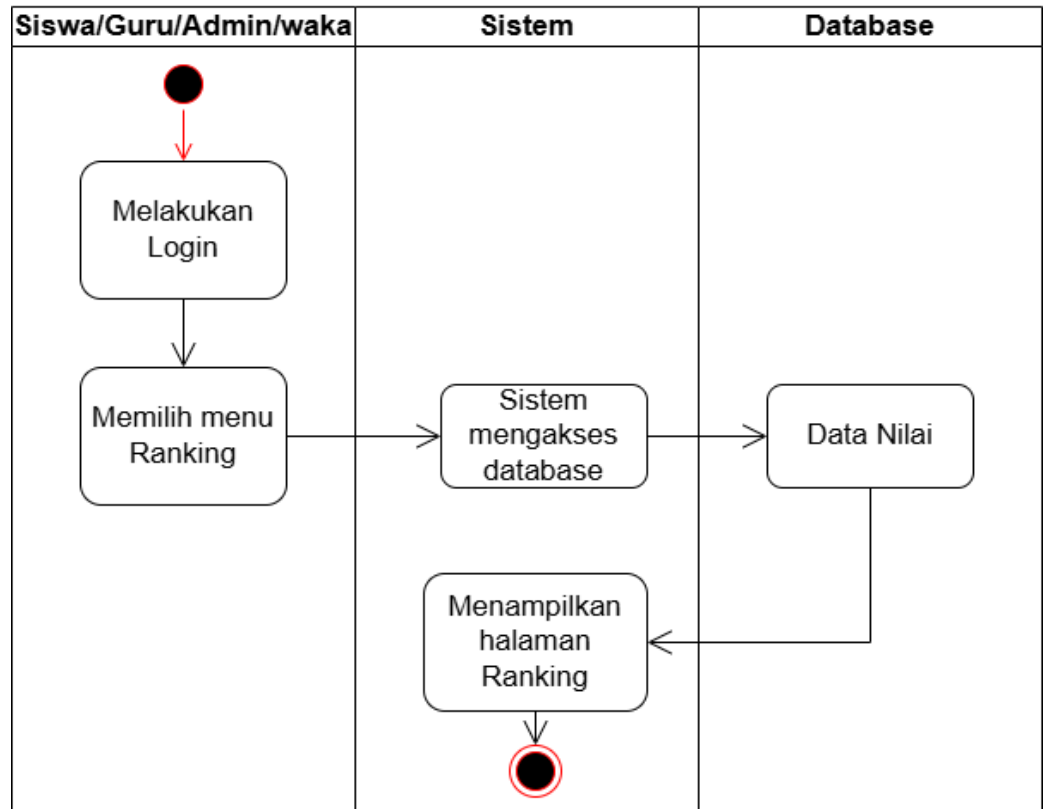


Gambar 22. Activity Diagram Melihat Ekstrakurikuler

8. Activity Diagram Melihat Ranking

Activity diagram yang ditunjukkan pada Gambar 23 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat *ranking* di dalam sistem. Proses dimulai ketika pengguna melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, pengguna memilih menu "*Ranking*" yang tersedia di sistem. Sistem kemudian mengakses *database* untuk mengambil data nilai yang tersimpan. Setelah data berhasil diambil, sistem akan menampilkan halaman yang berisi informasi *ranking* berdasarkan nilai siswa. Proses ini memastikan bahwa *ranking* yang ditampilkan adalah berdasarkan data yang valid dan akurat, serta memungkinkan pengguna untuk melihat peringkat siswa dalam konteks nilai yang diperoleh.

Activity Diagram Melihat Ranking



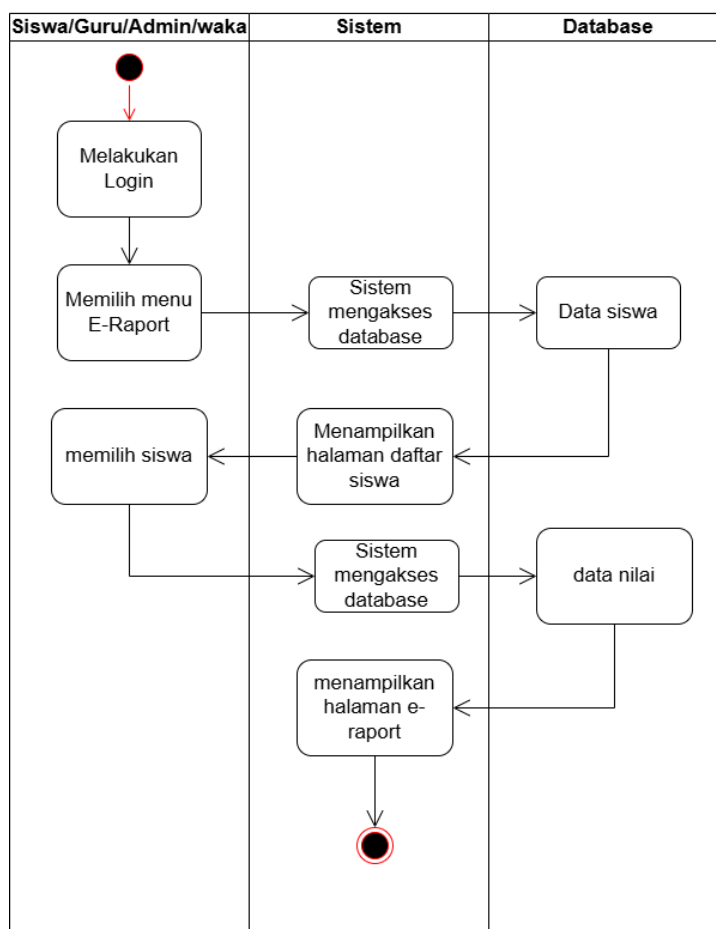
Gambar 23. Activity Diagram Melihat Ranking

9. Activity Diagram Melihat E-Raport

Activity diagram yang ditunjukkan pada Gambar 24 menjelaskan alur proses pengguna (siswa, guru, admin atau waka) dalam melihat *E-Raport* di dalam sistem. Proses dimulai dengan pengguna melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, pengguna memilih menu "*E-Raport*" yang tersedia di sistem. Sistem kemudian mengakses *database* untuk mengambil data siswa yang tersimpan. Setelah data berhasil diambil, sistem akan menampilkan halaman yang berisi daftar siswa. Dilanjutkan dengan pengguna memilih siswa. Sistem kemudian mengakses

database untuk mengambil data nilai yang tersimpan. Setelah data berhasil diambil, sistem akan menampilkan halaman e-raport yang berisi informasi terkait hasil belajar yang telah dicapai. Proses ini memastikan bahwa informasi *E-Raport* yang ditampilkan adalah akurat dan selalu diperbarui sesuai dengan data yang ada di *database*.

Activity Diagram Melihat E-Raport

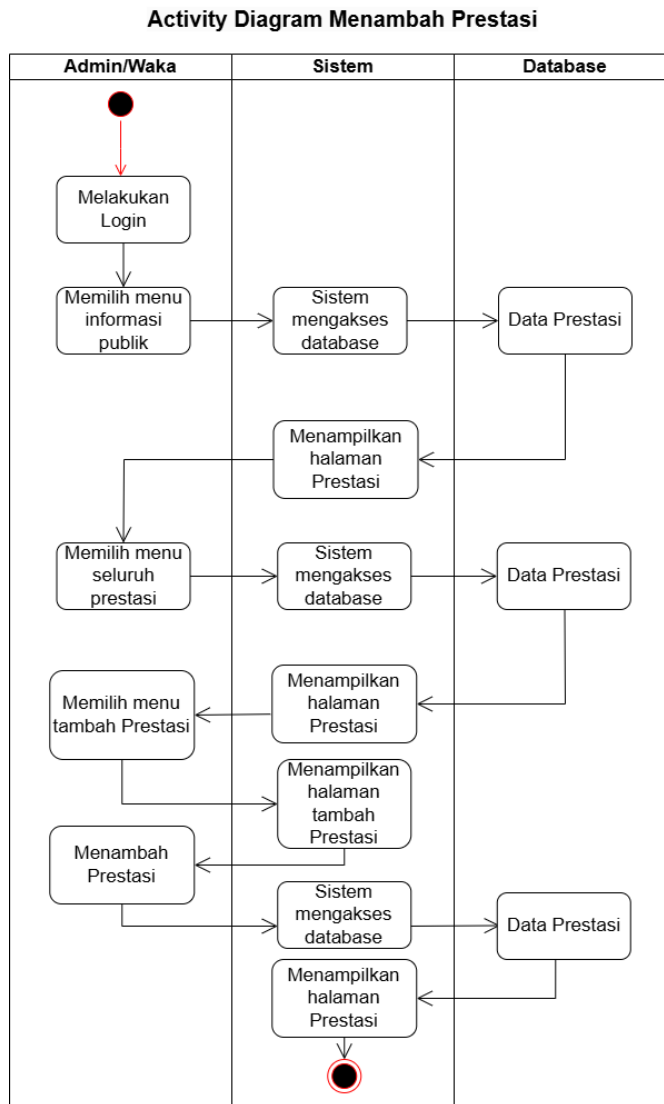


Gambar 24. Activity Diagram Melihat E-Raport

10. Activity Diagram Menambah Prestasi

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menambahkan data prestasi ke dalam sistem. Proses dimulai dengan admin melakukan *login*

untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses database untuk mengambil data prestasi yang telah tersimpan dan menampilkan halaman data prestasi. Admin kemudian memilih menu seluruh prestasi, sehingga sistem mengakses database kembali dan menampilkan halaman data prestasi. Selanjutnya, admin memilih menu tambah prestasi, sehingga sistem menampilkan halaman formulir untuk menambahkan data prestasi baru. Setelah admin mengisi data prestasi yang diperlukan, sistem mengakses database untuk menyimpan data yang telah dimasukkan. Setelah data berhasil disimpan, sistem menampilkan halaman prestasi yang telah diperbarui. Proses ini memastikan bahwa data prestasi dapat dikelola dengan mudah dan selalu diperbarui dengan informasi yang akurat dalam sistem.

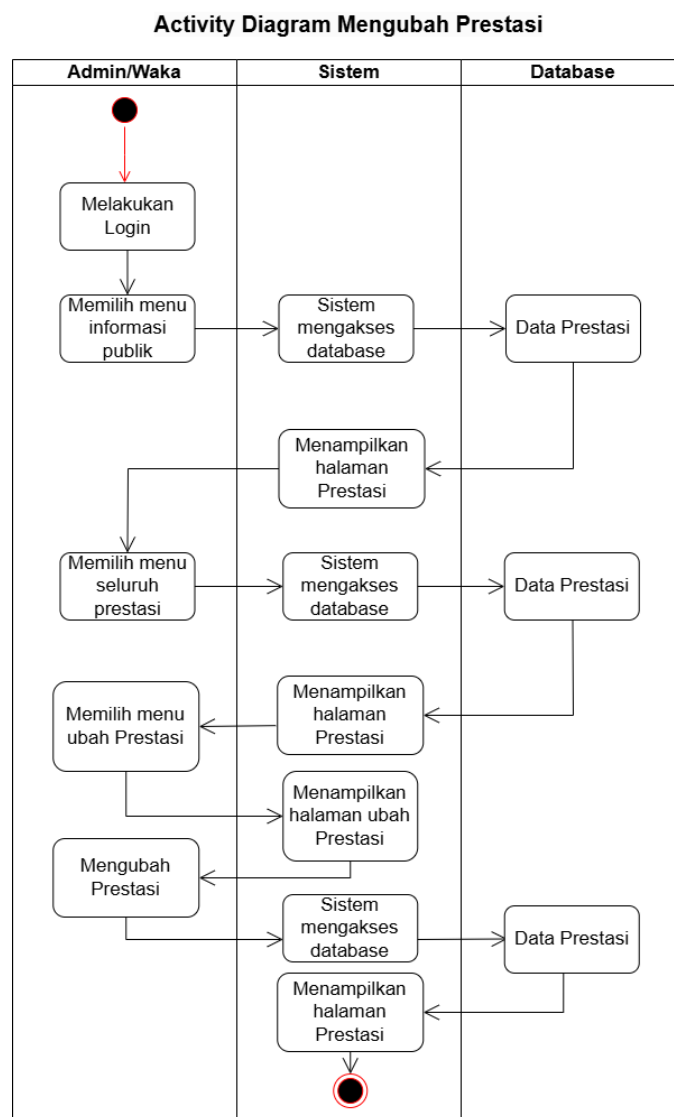


Gambar 25. Activity Diagram Menambah Prestasi

11. Activity Diagram Mengubah Prestasi

Activity diagram ini menjelaskan alur proses admin/waka dalam mengubah data prestasi di dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data prestasi yang telah tersimpan dan menampilkan halaman data prestasi. Admin kemudian memilih menu seluruh prestasi,

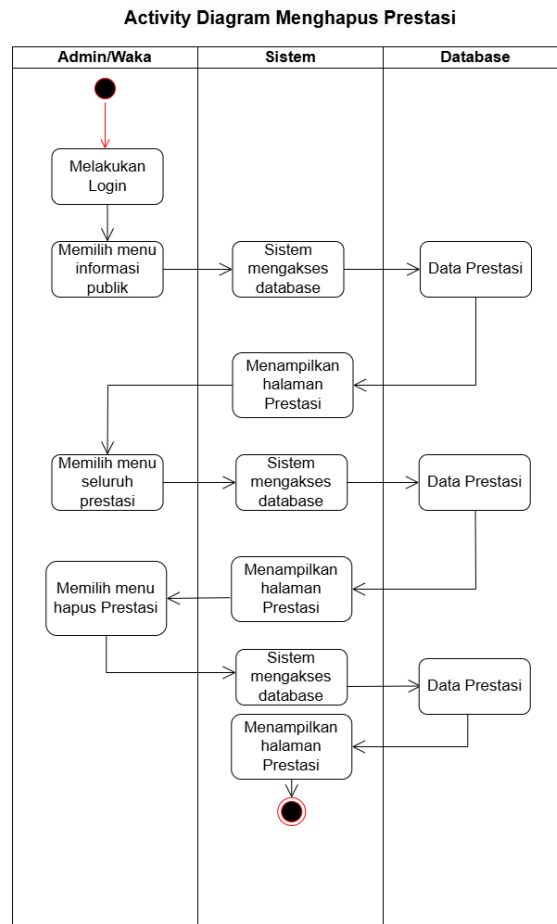
sehingga sistem kembali mengakses *database* dan menampilkan halaman data prestasi secara lengkap. Selanjutnya, admin memilih menu ubah prestasi, sehingga sistem menampilkan halaman formulir untuk mengedit data prestasi yang ada. Setelah admin melakukan perubahan pada data prestasi dan menyimpannya, sistem mengakses *database* untuk memperbarui data prestasi tersebut. Setelah data berhasil diperbarui, sistem menampilkan kembali halaman prestasi yang telah diperbarui.



Gambar 26. Activity Diagram Mengubah Prestasi

12. *Activity Diagram* Menghapus Prestasi

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menghapus data prestasi di dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data prestasi yang telah tersimpan dan menampilkan halaman data prestasi. Admin kemudian memilih menu seluruh prestasi, sehingga sistem mengakses *database* kembali dan menampilkan halaman data prestasi secara lengkap. Selanjutnya, admin memilih menu hapus prestasi, dan sistem mengakses *database* untuk menghapus data prestasi yang dipilih oleh admin. Setelah proses penghapusan berhasil, sistem menampilkan halaman prestasi yang telah diperbarui.



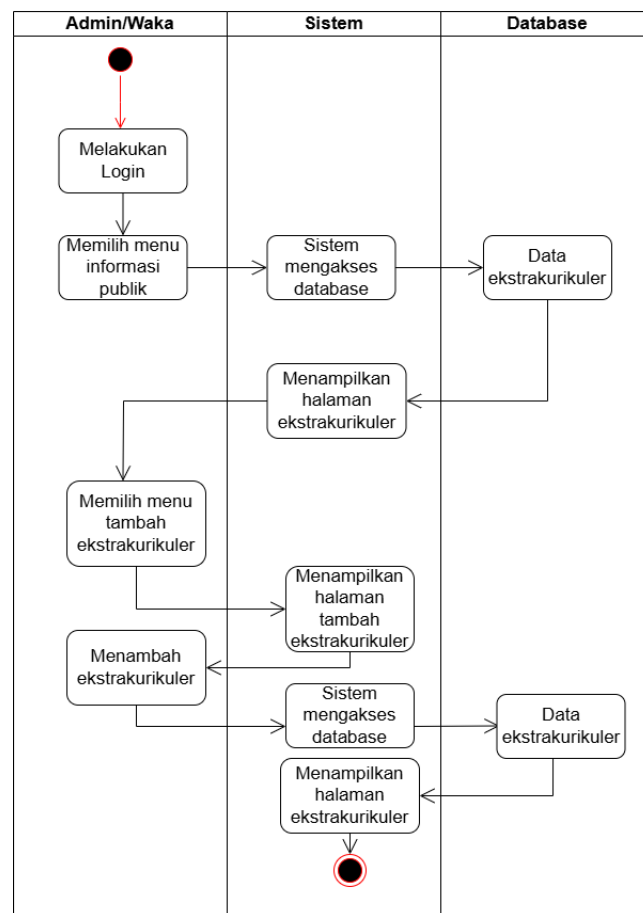
Gambar 27. *Activity Diagram Menghapus Prestasi*

13. *Activity Diagram Menambah Ekstrakurikuler*

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menambahkan data ekstrakurikuler ke dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data ekstrakurikuler yang telah tersimpan dan menampilkan halaman data ekstrakurikuler tersebut. Selanjutnya, admin memilih menu tambah ekstrakurikuler, sehingga sistem menampilkan halaman formulir untuk menambahkan data

ekstrakurikuler baru. Setelah admin mengisi data yang diperlukan dan mengirimkannya, sistem kembali mengakses *database* untuk menyimpan informasi yang baru ditambahkan. Setelah data berhasil disimpan, sistem menampilkan halaman ekstrakurikuler yang telah diperbarui. Proses ini memastikan bahwa data ekstrakurikuler dapat dikelola dengan mudah dan selalu diperbarui dengan informasi yang akurat dalam sistem.

Activity Diagram Menambah Ekstrakurikuler



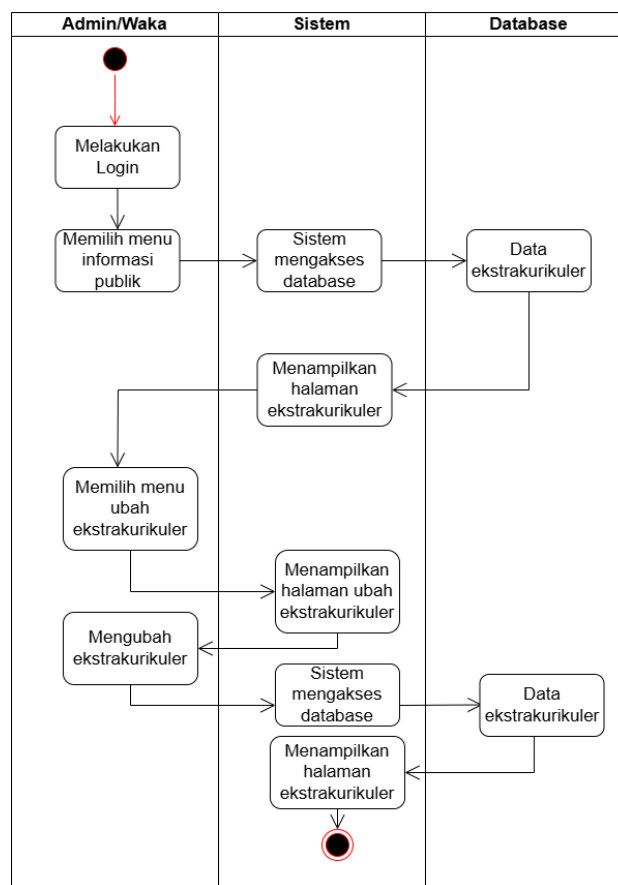
Gambar 28. *Activity Diagram Menambah Ekstrakurikuler*

14. *Activity Diagram Mengubah Ekstrakurikuler*

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam mengubah data ekstrakurikuler

di dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data ekstrakurikuler yang telah tersimpan dan menampilkan halaman data ekstrakurikuler. Selanjutnya, admin memilih menu ubah ekstrakurikuler, sehingga sistem menampilkan halaman formulir untuk mengedit data ekstrakurikuler yang ada. Setelah admin melakukan perubahan dan menyimpan data, sistem mengakses *database* untuk memperbarui data ekstrakurikuler tersebut. Setelah data berhasil diperbarui, sistem menampilkan halaman ekstrakurikuler yang telah diperbarui

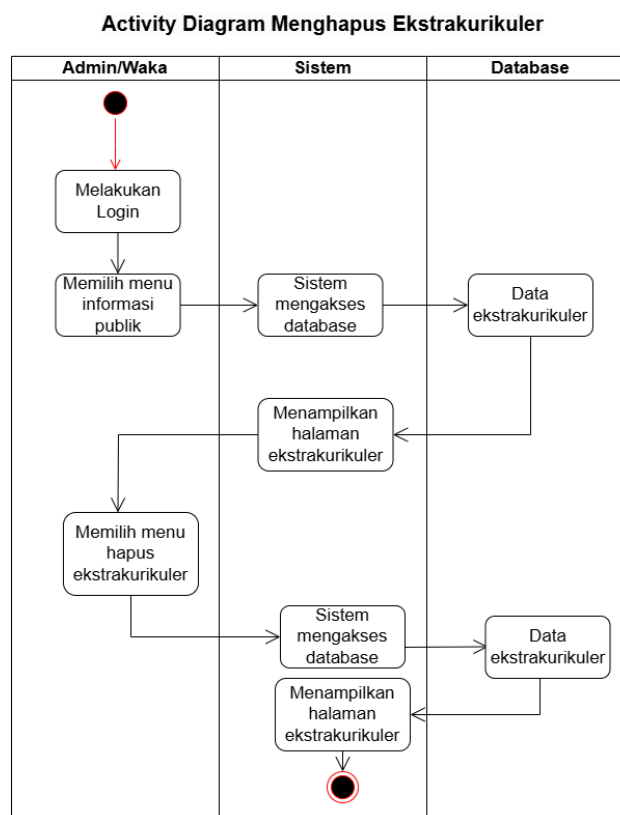
Activity Diagram Mengubah Ekstrakurikuler



Gambar 29. Activity Diagram Mengubah Ekstrakurikuler

15. Activity Diagram Menghapus Ekstrakurikuler

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menghapus data ekstrakurikuler di dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data ekstrakurikuler yang telah tersimpan dan menampilkan halaman data ekstrakurikuler. Selanjutnya, admin memilih menu hapus ekstrakurikuler. Setelah admin menghapus data ekstrakurikuler yang dipilih, sistem mengakses *database* untuk menghapus data tersebut. Setelah proses penghapusan selesai, sistem menampilkan halaman ekstrakurikuler yang telah diperbarui.

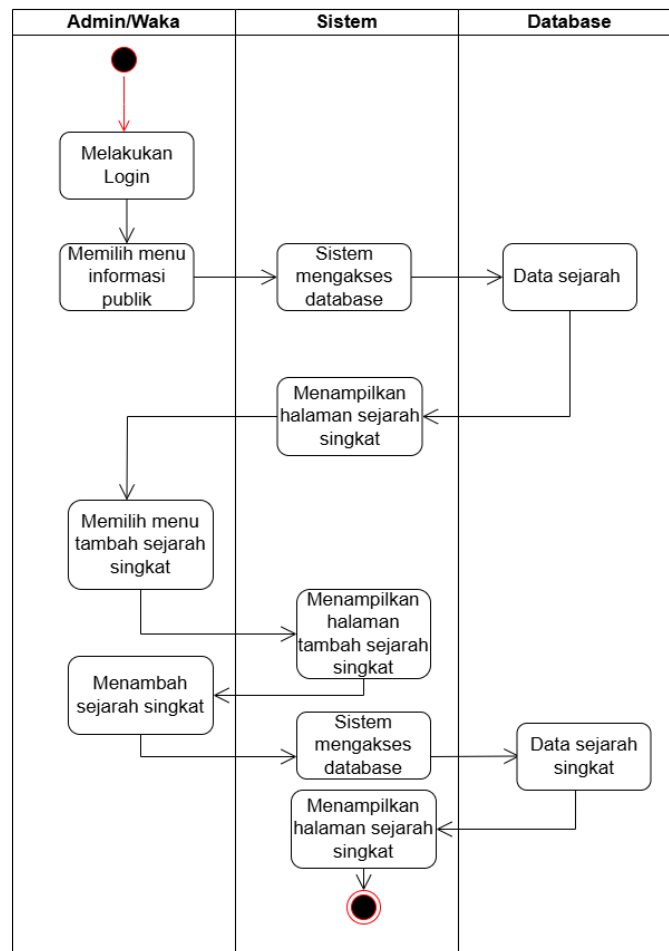


Gambar 30. Activity Diagram Menghapus Ekstrakurikuler

16. *Activity Diagram* Menambah Sejarah Singkat

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menambahkan data sejarah singkat ke dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data sejarah yang telah tersimpan dan menampilkan halaman sejarah singkat. Selanjutnya, admin memilih menu tambah sejarah singkat, sehingga sistem menampilkan halaman formulir untuk menambahkan data sejarah baru. Setelah admin mengisi data sejarah singkat dan menyimpannya, sistem mengakses *database* untuk menyimpan data yang baru dimasukkan. Setelah data berhasil disimpan, sistem menampilkan halaman sejarah singkat yang telah diperbarui. Proses ini memastikan bahwa data sejarah singkat dapat dikelola dengan mudah dan selalu diperbarui dengan informasi yang akurat dalam sistem.

Activity Diagram Menambah Sejarah Singkat



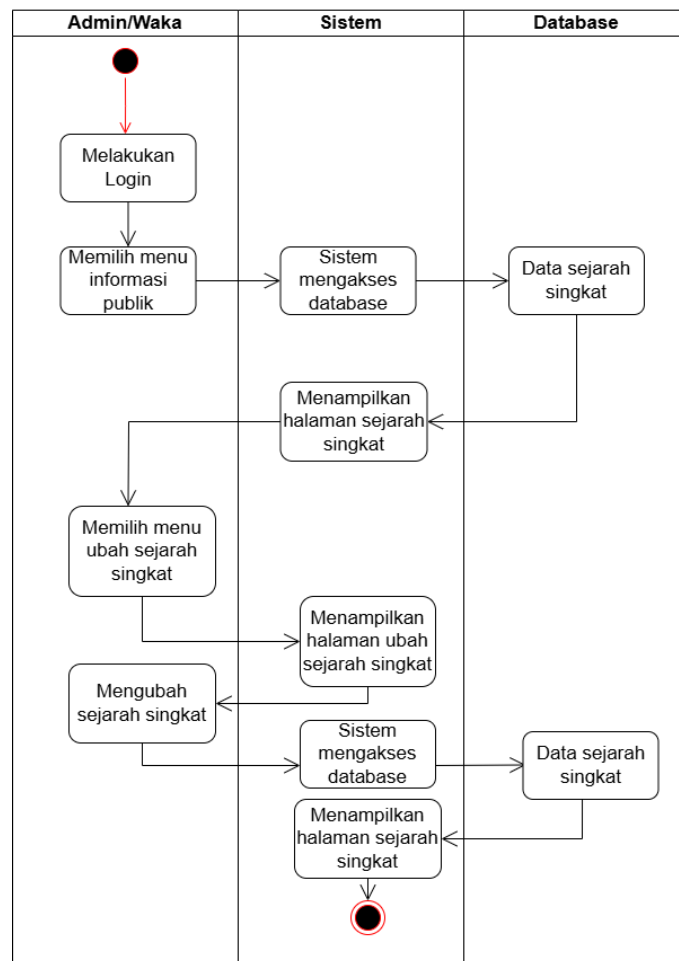
Gambar 31. Activity Diagram Menambah Sejarah Singkat

17. Activity Diagram Mengubah Sejarah Singkat

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam mengubah data sejarah singkat di dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data sejarah yang telah tersimpan dan menampilkan halaman sejarah singkat. Selanjutnya, admin memilih menu ubah sejarah singkat, sehingga sistem menampilkan halaman formulir untuk mengedit data sejarah

yang ada. Setelah admin melakukan perubahan dan menyimpan data tersebut, sistem mengakses *database* untuk memperbarui data sejarah. Setelah data berhasil diperbarui, sistem menampilkan halaman sejarah singkat yang telah diperbarui. Proses ini memastikan bahwa data sejarah singkat dapat dikelola dengan mudah dan selalu diperbarui dengan informasi yang akurat dalam sistem.

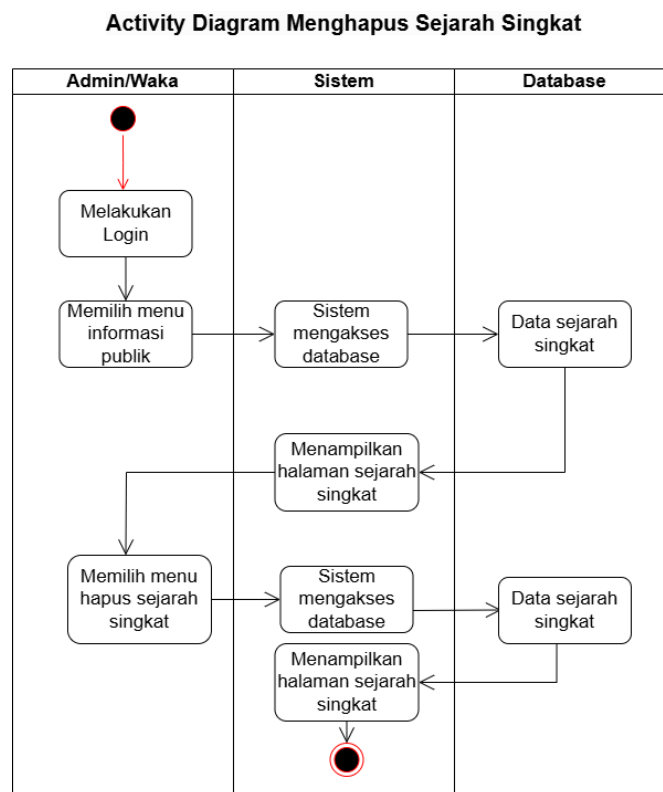
Activity Diagram Mengubah Sejarah Singkat



Gambar 32. Activity Diagram Mengubah Sejarah Singkat

18. Activity Diagram Menghapus Sejarah Singkat

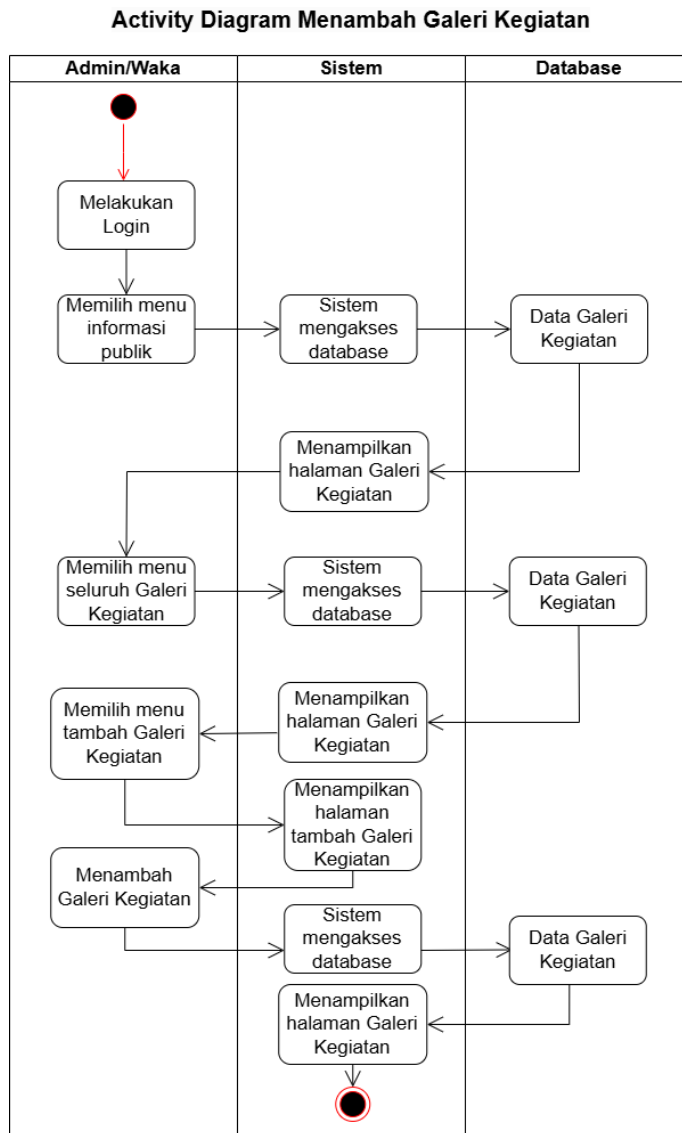
Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menghapus data sejarah singkat di dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data sejarah yang telah tersimpan dan menampilkan halaman sejarah singkat. Selanjutnya, admin memilih menu hapus sejarah singkat, sehingga sistem mengakses *database* untuk menghapus data sejarah yang dipilih. Setelah proses penghapusan selesai, sistem menampilkan halaman sejarah singkat yang telah diperbarui. Proses ini memastikan bahwa data sejarah singkat yang tidak diperlukan dapat dihapus dengan mudah, menjaga keteraturan serta keakuratan informasi dalam sistem.



Gambar 33. Activity Diagram Menghapus Sejarah Singkat

19. *Activity Diagram* Menambah Galeri Kegiatan

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menambahkan data galeri kegiatan ke dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data galeri kegiatan yang telah tersimpan dan menampilkan halaman galeri kegiatan. Selanjutnya, admin memilih menu seluruh galeri kegiatan, sehingga sistem kembali mengakses *database* dan menampilkan halaman galeri kegiatan secara lengkap. Admin kemudian memilih menu tambah galeri kegiatan, sehingga sistem menampilkan halaman formulir untuk menambahkan data galeri kegiatan baru. Setelah admin mengisi data yang diperlukan dan mengirimkannya, sistem mengakses *database* untuk menyimpan informasi yang baru ditambahkan. Setelah data berhasil disimpan, sistem menampilkan halaman galeri kegiatan yang telah diperbarui. Proses ini memastikan bahwa data galeri kegiatan dapat dikelola dengan mudah dan selalu diperbarui dengan informasi yang akurat dalam sistem.

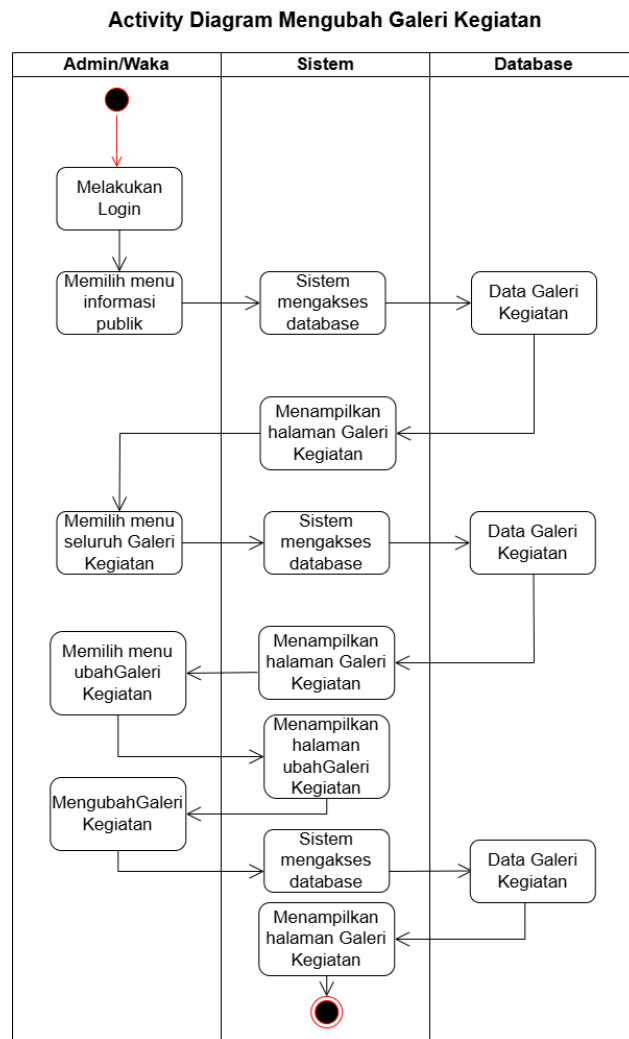


Gambar 34. *Activity Diagram* Menambah Galeri Kegiatan

20. *Activity Diagram* Mengubah Galeri Kegiatan

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam mengubah data galeri kegiatan di dalam sistem. Proses dimulai dengan admin melakukan login untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data galeri kegiatan yang telah

tersimpan dan menampilkan halaman galeri kegiatan. Selanjutnya, admin memilih menu seluruh galeri kegiatan, sehingga sistem kembali mengakses *database* dan menampilkan halaman galeri kegiatan secara lengkap. Admin kemudian memilih menu ubah galeri kegiatan, sehingga sistem menampilkan halaman formulir untuk mengedit data galeri kegiatan yang ada. Setelah admin melakukan perubahan dan menyimpan data tersebut, sistem mengakses *database* untuk memperbarui data galeri kegiatan. Setelah data berhasil diperbarui, sistem menampilkan halaman galeri kegiatan yang telah diperbarui. Proses ini memastikan bahwa data galeri kegiatan dapat dikelola dengan mudah dan selalu diperbarui dengan informasi yang akurat dalam sistem.

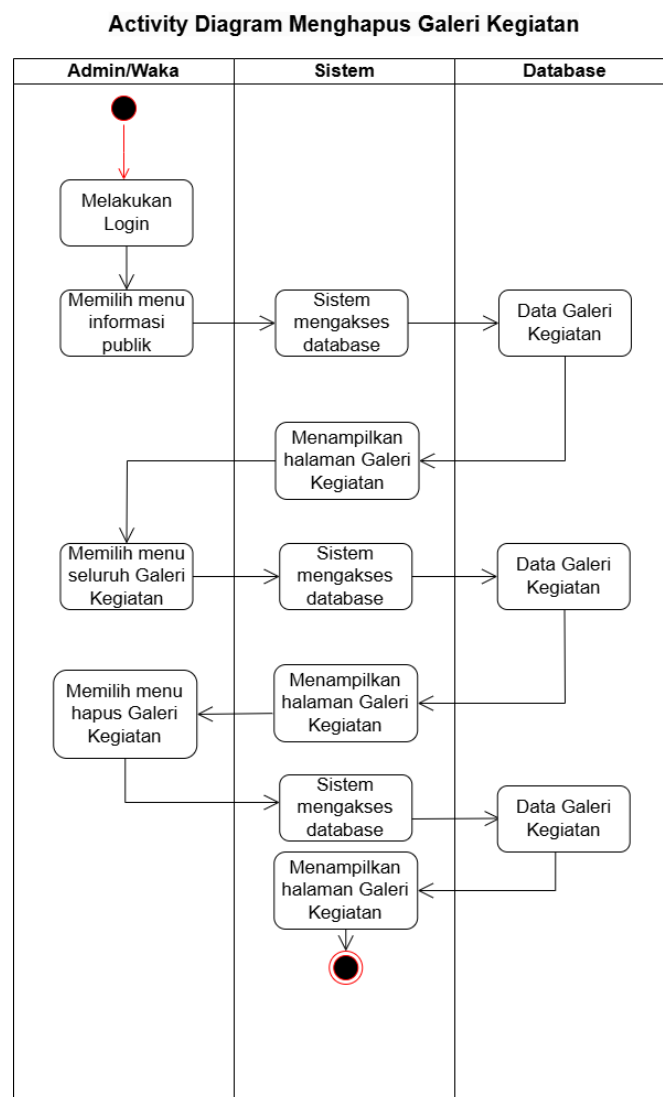


Gambar 35. *Activity Diagram* Mengubah Galeri Kegiatan

21. *Activity Diagram* Menghapus Galeri Kegiatan

Activity diagram yang ditunjukkan pada gambar menjelaskan alur proses admin/waka dalam menghapus data galeri kegiatan di dalam sistem. Proses dimulai dengan admin melakukan *login* untuk mendapatkan akses. Setelah *login* berhasil, admin memilih menu informasi publik. Sistem kemudian mengakses *database* untuk mengambil data galeri kegiatan yang telah tersimpan dan menampilkan halaman galeri kegiatan. Selanjutnya, admin memilih menu seluruh galeri kegiatan,

sehingga sistem kembali mengakses *database* dan menampilkan halaman galeri kegiatan secara lengkap. Admin kemudian memilih menu hapus galeri kegiatan, sehingga sistem mengakses *database* untuk menghapus data galeri kegiatan yang dipilih. Setelah proses penghapusan selesai, sistem menampilkan halaman galeri kegiatan yang telah diperbarui. Proses ini memastikan bahwa data galeri kegiatan yang tidak diperlukan dapat dihapus dengan mudah, menjaga keteraturan serta keakuratan informasi dalam sistem.



Gambar 36. Activity Diagram Menghapus Galeri Kegiatan

E. Use Case Description

Deskripsi setiap *Use Case Diagram* adalah sebagai berikut:

Tabel 5. Deskripsi *Use Case* untuk Melihat Visi Misi

Use Case Name: Melihat Visi Misi	ID: UC-01
Actor: Siswa, Guru, Waka dan Admin	
Preconditions: Siswa, Guru, Waka dan Admin tidak harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin memilih menu visi misi 2. Sistem menampilkan halaman visi misi	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman visi misi	

Tabel 6. Deskripsi *Use Case* untuk Melihat Struktur Organisasi

Use Case Name: Melihat Struktur Organisasi	ID: UC-02
Actor: Siswa, Guru, Waka dan Admin	
Preconditions Siswa, Guru, Waka dan Admin tidak harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin memilih menu struktur organisasi 2. Sistem menampilkan halaman struktur organisasi	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman struktur organisasi	

Tabel 7. Deskripsi *Use Case* untuk Melihat Hymne

Use Case Name: Melihat Hymne	ID: UC-03
Actor: Siswa, Guru, Waka dan Admin	
Preconditions Siswa, Guru, Waka dan Admin tidak harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin memilih menu hymne 2. Sistem menampilkan halaman hymne	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman hymne	

Tabel 8. Deskripsi *Use Case* untuk Melihat Sejarah Singkat

Use Case Name: Melihat Sejarah Singkat	ID: UC-04
Actor: Siswa, Guru, Waka dan Admin	
Preconditions: Siswa, Guru, Waka dan Admin tidak harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin memilih menu sejarah singkat 2. Sistem menampilkan halaman sejarah singkat	

Postconditions:

Siswa, Guru, Waka dan Admin berhasil membuka halaman sejarah singkat

Tabel 9. Deskripsi *Use Case* untuk Melihat Prestasi

Use Case Name: Melihat Prestasi	ID: UC-05
Actor: Siswa, Guru, Waka dan Admin	
Preconditions: Siswa, Guru, Waka dan Admin tidak harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin memilih menu prestasi 2. Sistem meminta data prestasi kepada <i>database</i> 3. <i>Database</i> memberikan data prestasi kepada sistem 4. Sistem menampilkan halaman prestasi	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman prestasi	

Tabel 10. Deskripsi *Use Case* untuk Melihat Galeri Kegiatan

Use Case Name: Melihat Galeri Kegiatan	ID: UC-06
Actor: Siswa, Guru, Waka dan Admin	
Preconditions: Siswa, Guru, Waka dan Admin tidak harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin memilih menu galeri kegiatan 2. Sistem meminta data galeri kegiatan kepada <i>database</i> 3. <i>Database</i> memberikan data galeri kegiatan kepada sistem 4. Sistem menampilkan halaman galeri kegiatan	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman galeri kegiatan	

Tabel 11. Deskripsi *Use Case* untuk Melihat Ekstrakurikuler

Use Case Name: Melihat Ekstrakurikuler	ID: UC-07
Actor: Siswa, Guru, Waka dan Admin	
Preconditions: Siswa, Guru, Waka dan Admin tidak harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin memilih menu ekstrakurikuler 2. Sistem meminta data ekstrakurikuler kepada <i>database</i> 3. <i>Database</i> memberikan data ekstrakurikuler kepada sistem 4. Sistem menampilkan halaman ekstrakurikuler	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman ekstrakurikuler	

Tabel 12. Deskripsi *Use Case* untuk Melihat *Ranking*

Use Case Name: Melihat <i>Ranking</i>	ID: UC-08
Actor: Siswa, Guru, Waka dan Admin	
Preconditions: Siswa, Guru, Waka dan Admin harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin melakukan <i>login</i> 2. Siswa, Guru, Waka dan Admin memilih menu <i>ranking</i> 3. Sistem meminta data nilai kepada <i>database</i> 4. <i>Database</i> memberikan data nilai kepada sistem 5. Sistem menampilkan halaman <i>ranking</i>	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman <i>ranking</i>	

Tabel 13. Deskripsi *Use Case* untuk Melihat *E-Raport*

Use Case Name: Melihat <i>E-Raport</i>	ID: UC-09
Actor: Siswa, Guru, Waka dan Admin	
Preconditions: Siswa, Guru, Waka dan Admin harus <i>login</i> kedalam sistem	
Normal Course: 1. Siswa, Guru, Waka dan Admin melakukan <i>login</i> 2. Siswa, Guru, Waka dan Admin memilih menu <i>E-Raport</i> 3. Sistem meminta data nilai kepada <i>database</i> 4. <i>Database</i> memberikan data nilai kepada sistem 5. Sistem menampilkan halaman <i>E-Raport</i>	
Postconditions: Siswa, Guru, Waka dan Admin berhasil membuka halaman <i>E-Raport</i>	

Tabel 14. Deskripsi *Use Case* untuk Menambah Prestasi

Use Case Name: Menambah Prestasi	ID: UC-10
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus sudah <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu prestasi 3. Sistem meminta data kepada <i>database</i> 4. <i>Database</i> memberikan hasil pencarian prestasi kepada sistem 5. Sistem menampilkan halaman prestasi 6. Admin dan Waka memilih menu tambah prestasi 7. Sistem menampilkan halaman tambah prestasi 8. Admin dan Waka menambah prestasi 9. Sistem mengirim data prestasi kepada <i>database</i> 10. <i>Database</i> menyimpan prestasi 11. Sistem menampilkan halaman prestasi	

Postconditions:

Admin dan Waka berhasil menambah prestasi

Tabel 15. Deskripsi *Use Case* untuk Mengubah Prestasi

Use Case Name: Mengubah Prestasi	ID: UC-11
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus sudah <i>login</i> kedalam sistem	
Normal Course: 12. Admin dan Waka melakukan <i>login</i> 13. Admin dan Waka memilih menu prestasi 14. Sistem meminta data kepada <i>database</i> 15. <i>Database</i> memberikan hasil pencarian prestasi kepada sistem 16. Sistem menampilkan halaman prestasi 17. Admin dan Waka memilih menu ubah prestasi 18. Sistem menampilkan halaman ubah prestasi 19. Admin dan Waka mengubah prestasi 20. Sistem mengirim data prestasi kepada <i>database</i> 21. <i>Database</i> menyimpan data prestasi 22. Sistem menampilkan halaman prestasi	
Postconditions: Admin dan Waka berhasil mengubah prestasi	

Tabel 16. Deskripsi *Use Case* untuk Menghapus Prestasi

Use Case Name: Menghapus Prestasi	ID: UC-12
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus sudah <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu Prestasi 3. Sistem meminta data kepada <i>database</i> 4. <i>Database</i> memberikan hasil pencarian Prestasi kepada sistem 5. Sistem menampilkan halaman Prestasi 6. Admin dan Waka memilih menu hapus Prestasi 7. Sistem mengirim perintah hapus Prestasi kepada <i>database</i> 8. <i>Database</i> menghapus Prestasi 9. Sistem menampilkan halaman Prestasi	
Postconditions: Admin dan Waka berhasil menghapus Prestasi	

Tabel 17. Deskripsi *Use Case* untuk Menambah Ekstrakurikuler

Use Case Name: Menambah Ekstrakurikuler	ID: UC-13
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu ekstrakurikuler 3. Sistem meminta data ekstrakurikuler kepada <i>database</i> 4. <i>Database</i> memberikan data ekstrakurikuler kepada sistem 5. Sistem menampilkan halaman ekstrakurikuler 6. Admin dan Waka memilih menu tambah ekstrakurikuler 7. Sistem menampilkan halaman tambah ekstrakurikuler 8. Admin dan Waka mengisi form tambah ekstrakurikuler 9. Sistem mengirim data ekstrakurikuler ke <i>database</i> 10. <i>Database</i> menyimpan data ekstrakurikuler 11. Sistem menampilkan halaman ekstrakurikuler	
Postconditions: Admin dan Waka berhasil menambahkan ekstrakurikuler	

Tabel 18. Deskripsi *Use Case* untuk Mengubah Ekstrakurikuler

Use Case Name: Mengubah Ekstrakurikuler	ID: UC-14
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu ekstrakurikuler 3. Sistem meminta data ekstrakurikuler kepada <i>database</i> 4. <i>Database</i> memberikan data ekstrakurikuler kepada sistem 5. Sistem menampilkan halaman ekstrakurikuler 6. Admin dan Waka memilih menu ubah ekstrakurikuler 7. Sistem menampilkan halaman ubah ekstrakurikuler 8. Admin dan Waka mengisi form ubah ekstrakurikuler 9. Sistem mengirim data ekstrakurikuler ke <i>database</i> 10. <i>Database</i> menyimpan data ekstrakurikuler 11. Sistem menampilkan halaman ekstrakurikuler	
Postconditions: Admin dan Waka berhasil mengubah ekstrakurikuler	

Tabel 19. Deskripsi *Use Case* untuk Menghapus Ekstrakurikuler

Use Case Name: Menghapus Ekstrakurikuler	ID: UC-15
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	

Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu ekstrakurikuler 3. Sistem meminta data ekstrakurikuler kepada <i>database</i> 4. <i>Database</i> memberikan data ekstrakurikuler kepada sistem 5. Sistem menampilkan halaman ekstrakurikuler 6. Admin dan Waka memilih menu hapus ekstrakurikuler 7. Sistem mengirim perintah hapus ke <i>database</i> 8. <i>Database</i> menghapus data ekstrakurikuler 9. Sistem menampilkan halaman ekstrakurikuler
Postconditions: Admin dan Waka berhasil menghapus ekstrakurikuler

Tabel 20. Deskripsi *Use Case* untuk Menambah Sejarah Singkat

Use Case Name: Menambah Sejarah Singkat	ID: UC-16
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu sejarah singkat 3. Sistem meminta data sejarah singkat kepada <i>database</i> 4. <i>Database</i> memberikan data sejarah singkat kepada sistem 5. Sistem menampilkan halaman sejarah singkat 6. Admin dan Waka memilih menu tambah sejarah singkat 7. Sistem menampilkan halaman tambah sejarah singkat 8. Admin dan Waka mengisi form tambah sejarah singkat 9. Sistem mengirim data sejarah singkat ke <i>database</i> 10. <i>Database</i> menyimpan data sejarah singkat 11. Sistem menampilkan halaman sejarah singkat	
Postconditions: Admin dan Waka berhasil menambahkan sejarah singkat	

Tabel 21. Deskripsi *Use Case* untuk Mengubah Sejarah Singkat

Use Case Name: Mengubah Sejarah Singkat	ID: UC-17
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu sejarah singkat 3. Sistem meminta data sejarah singkat kepada <i>database</i> 4. <i>Database</i> memberikan data sejarah singkat kepada sistem 5. Sistem menampilkan halaman sejarah singkat 6. Admin dan Waka memilih menu ubah sejarah singkat	

7. Sistem menampilkan halaman ubah sejarah singkat 8. Admin dan Waka mengisi form ubah sejarah singkat 9. Sistem mengirim data sejarah singkat ke <i>database</i> 10. <i>Database</i> menyimpan data sejarah singkat 11. Sistem menampilkan halaman sejarah singkat
Postconditions: Admin dan Waka berhasil mengubah sejarah singkat

Tabel 22. Deskripsi *Use Case* untuk Menghapus Sejarah Singkat

Use Case Name: Menghapus Sejarah Singkat	ID: UC-18
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu sejarah singkat 3. Sistem meminta data sejarah singkat kepada <i>database</i> 4. <i>Database</i> memberikan data sejarah singkat kepada sistem 5. Sistem menampilkan halaman sejarah singkat 6. Admin dan Waka memilih menu hapus sejarah singkat 7. Sistem mengirim perintah hapus ke <i>database</i> 8. <i>Database</i> menghapus data sejarah singkat 9. Sistem menampilkan halaman sejarah singkat	
Postconditions: Admin dan Waka berhasil menghapus sejarah singkat	

Tabel 23. Deskripsi *Use Case* untuk Menambah Galeri Kegiatan

Use Case Name: Menambah Galeri Kegiatan	ID: UC-19
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu galeri kegiatan 3. Sistem meminta data galeri kegiatan kepada <i>database</i> 4. <i>Database</i> memberikan data galeri kegiatan kepada sistem 5. Sistem menampilkan halaman galeri kegiatan 6. Admin dan Waka memilih menu tambah galeri kegiatan 7. Sistem menampilkan halaman tambah galeri kegiatan 8. Admin dan Waka mengisi form tambah galeri kegiatan 9. Sistem mengirim data galeri kegiatan ke <i>database</i> 10. <i>Database</i> menyimpan data galeri kegiatan 11. Sistem menampilkan halaman galeri kegiatan	
Postconditions: Admin dan Waka berhasil menambahkan galeri kegiatan	

Tabel 24. Deskripsi *Use Case* untuk Mengubah Galeri Kegiatan

Use Case Name: Mengubah Galeri Kegiatan	ID: UC-20
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu galeri kegiatan 3. Sistem meminta data galeri kegiatan kepada <i>database</i> 4. <i>Database</i> memberikan data galeri kegiatan kepada sistem 5. Sistem menampilkan halaman galeri kegiatan 6. Admin dan Waka memilih menu ubah galeri kegiatan 7. Sistem menampilkan halaman ubah galeri kegiatan 8. Admin dan Waka mengisi form ubah galeri kegiatan 9. Sistem mengirim data galeri kegiatan ke <i>database</i> 10. <i>Database</i> menyimpan data galeri kegiatan 11. Sistem menampilkan halaman galeri kegiatan	
Postconditions: Admin dan Waka berhasil mengubah galeri kegiatan	

Tabel 25. Deskripsi *Use Case* untuk Menghapus Galeri Kegiatan

Use Case Name: Menghapus Galeri Kegiatan	ID: UC-21
Actor: Admin dan Waka	
Preconditions: Admin dan Waka harus <i>login</i> kedalam sistem	
Normal Course: 1. Admin dan Waka melakukan <i>login</i> 2. Admin dan Waka memilih menu galeri kegiatan 3. Sistem meminta data galeri kegiatan kepada <i>database</i> 4. <i>Database</i> memberikan data galeri kegiatan kepada sistem 5. Sistem menampilkan halaman galeri kegiatan 6. Admin dan Waka memilih menu hapus galeri kegiatan 7. Sistem mengirim perintah hapus ke <i>database</i> 8. <i>Database</i> menghapus data galeri kegiatan 9. Sistem menampilkan halaman galeri kegiatan	
Postconditions: Admin dan Waka berhasil menghapus galeri kegiatan	

F. Strategi Analisis Kebutuhan

Pemetaan Kebutuhan dengan Proses *As-Is* dan *To-Be-System*

Berikut ini adalah pemetaan kebutuhan (baik fungsional maupun nonfungsional) dengan proses *As-Is* dan proses *To-Be*.

Tabel 26. Matriks Kebutuhan Fungsional As-Is dan To-Be

Kebutuhan Fungsional		
No.	Proses berjalan (<i>As-Is</i>)	Sistem yang akan dibuat (<i>To-Be</i>)
1.	Admin mengelola informasi publik (Sejarah singkat, galeri kegiatan, prestasi, ekstrakurikuler) secara manual menggunakan pengumuman sederhana, dokumen kertas atau diupdate di media sosial.	Admin dapat mengelola dan memperbarui informasi publik (Sejarah singkat, galeri kegiatan, prestasi, ekstrakurikuler) melalui sistem yang terintegrasi dan dapat diakses secara online.
2.	Siswa, guru dan admin mendapatkan informasi (Sejarah singkat, galeri kegiatan, prestasi, ekstrakurikuler, visi misi, hymne, struktur organisasi) hanya melalui pengumuman langsung atau informasi yang terbatas pada media sosial.	Siswa, guru, dan admin dapat mengakses informasi publik (Sejarah singkat, galeri kegiatan, prestasi, ekstrakurikuler, visi misi, hymne, struktur organisasi) kapan saja dan dari mana saja melalui <i>website</i> yang menyediakan informasi terstruktur dan transparan.
3.	Guru dan admin mencatat dan mengolah nilai secara manual menggunakan kertas atau spreadsheet, yang rentan terhadap kesalahan dan membutuhkan waktu lebih lama untuk pengolahan data.	Guru dan admin dapat langsung menginput, mengubah, atau menghapus nilai melalui sistem, guna mengurangi risiko kesalahan dan meningkatkan efisiensi.
4.	Siswa hanya dapat melihat nilai, <i>ranking</i> , kemajuan belajar dan <i>raport</i> pada saat penerimaan rapor akhir semester, dan tidak dapat	Siswa dapat memantau kemajuan belajar, nilai, <i>raport</i> , dan <i>ranking</i> siswa secara langsung dapat diakses dengan cepat melalui

	memantau hasil belajar secara <i>real-time</i>	sistem yang terintegrasi, memungkinkan pemantauan <i>real-time</i> , akurat dan transparansi yang lebih tinggi.
5.	Admin mengelola data akademik, seperti data siswa, data guru pegawai, jadwal pelajaran dan riwayat administrasi secara manual menggunakan dokumen fisik atau spreadsheet sederhana, yang rentan terhadap kesalahan dan memakan waktu.	Admin dapat mengelola data akademik (data siswa, data guru pegawai, jadwal pelajaran dan riwayat administrasi) secara terintegrasi dalam sistem berbasis <i>web</i> , yang memastikan data lebih aman, akurat, dan dapat diakses dengan cepat.
6.	Guru dan admin mengelola <i>ranking</i> , kemajuan belajar dan <i>raport</i> kelas secara manual yang memakan lebih banyak waktu	Guru dan admin bisa langsung melihat data kemajuan belajar, <i>ranking</i> dan <i>raport</i> , karena berdasarkan nilai, <i>ranking</i> kelas, <i>ranking</i> angkatan, kemajuan belajar dan data <i>raport</i> akan otomatis terhitung dan <i>ter-update</i> data nya secara <i>real-time</i>
7.	Siswa dan guru hanya dapat melihat data akademik seperti data siswa, guru, jadwal pelajaran dan riwayat administrasi secara terbatas, tidak ada sistem yang memungkinkan untuk mengakses data akademik kapan saja.	Siswa dan guru dapat melihat data akademik secara langsung melalui sistem berbasis <i>web</i> kapan saja dan di mana saja

Tabel 27. Matriks Kebutuhan Non-Fungsional As-Is dan To-Be

Kebutuhan Non-Fungsional		
No.	Proses berjalan (<i>As-Is</i>)	Sistem yang akan dibuat (<i>To-Be</i>)
1.	Persyaratan Operasional	
	- Sistem pengelolaan data akademik hanya tersedia dalam bentuk manual atau semi-digital, seperti menggunakan dokumen cetak atau aplikasi desktop lokal.	Sistem harus dapat diakses secara <i>online</i> baik melalui <i>website</i> .
	- Pengguna hanya dapat mengakses sistem di komputer yang tersedia di ruang administrasi sekolah, tanpa dukungan perangkat mobile atau akses <i>online</i> .	Sistem harus dapat dijalankan di perangkat <i>smartphone</i> , tablet, maupun komputer.
2.	Persyaratan Kinerja	
	- Proses pengelolaan nilai dan data akademik membutuhkan waktu lama karena dilakukan secara manual atau menggunakan perangkat lunak sederhana tanpa optimasi fitur.	Sistem harus mampu memproses penggunaan fitur, seperti akses data akademik dan unggahan nilai pembelajaran, dalam waktu tidak lebih dari 30 detik per permintaan.
3.	Persyaratan Keandalan	
	- Layanan tidak dapat diakses kapanpun, hanya dapat diakses saat jam kerja saja.	Sistem harus memiliki <i>uptime</i> 99.9% untuk memastikan ketersediaan layanan yang konsisten bagi pengguna.

4.	Persyaratan Keselamatan	
	- Belum ada Sistem informasi akademik di sekolah.	Enkripsi data harus diterapkan untuk semua data sensitif seperti <i>password</i> akun pengguna.
5.	Persyaratan Keamanan	
	- Otorisasi belum dilakukan oleh sistem, sehingga terbilang lebih rentan.	Sistem harus memastikan bahwa setiap fitur hanya dapat diakses oleh pengguna yang memiliki peran dan otorisasi sesuai dengan hak akses yang telah ditentukan.
6.	Persyaratan Kegunaan	
	- Proses masih dilakukan secara manual atau belum terdigitalisasi, sehingga belum ada antar muka sistem.	Antarmuka pengguna harus intuitif dan mudah digunakan, dengan desain yang responsif untuk berbagai perangkat.

3. Coding (Pengkodean)

Sistem informasi akademik (SIKAD) di SMPN 29 Bandar Lampung dikembangkan dengan menggunakan *framework* CodeIgniter 4 dan bahasa pemrograman PHP. Sistem ini terdiri dari empat modul utama, yakni Informasi Publik, Data Akademik, Pembelajaran, dan Evaluasi. Proses pengembangan dibagi berdasarkan modul, di mana saya bertanggung jawab pada modul Informasi Publik dan Evaluasi, sementara modul Data Akademik dan Pembelajaran dikembangkan oleh Muhammad Rafi Satria. Pembagian tugas tersebut bertujuan untuk mempercepat pengerjaan sekaligus memastikan setiap modul berfungsi dengan baik.

4. *Testing* (Pengujian)

Tahap pengujian merupakan langkah terakhir dalam metode Extreme Programming dan memiliki peran penting dalam memastikan kualitas sistem. Pada tahap ini, pengguna dilibatkan secara langsung untuk menilai apakah sistem yang dikembangkan telah memenuhi kebutuhan yang diharapkan. Dalam pelaksanaan pengujian pada penelitian ini, digunakan dua metode, yaitu Black Box Testing dan User Acceptance Testing (UAT). Black Box Testing difokuskan pada pengujian fungsional sistem dengan cara memberikan input dan mengamati output tanpa memperhatikan struktur internal program. Sedangkan UAT dilakukan dengan melibatkan pengguna untuk mengevaluasi apakah sistem telah berjalan sesuai dengan tujuan dan dapat diterima untuk digunakan dalam lingkungan sebenarnya. Skenario Pengujian Black Box dan UAT dapat dilihat pada tabel 28 dan 29.

Tabel 28. Skenario Black Box Testing

No.	Kelas Pengujian	ID Kasus Pengujian	Kasus Pengujian	Hasil Yang diharapkan
1.	Menu Sejarah Singkat	UC04-TC01	Menekan menu sejarah singkat	Menampilkan halaman sejarah singkat
		UC16-TC01	Menekan tombol tambah kepemimpinan	Menampilkan form tambah kepemimpinan
		UC16-TC02	Mengisi data kepemimpinan dengan input valid	Data berhasil disimpan dan menampilkan semua data kepemimpinan
		UC16-TC03	Mengisi form dengan data kosong	Sistem menampilkan pesan error validasi
		UC17-TC01	Menekan tombol edit kepemimpinan	Menampilkan form edit kepemimpinan

		UC17-TC02	Mengedit data dengan input valid	Data berhasil disimpan dan menampilkan semua data kepemimpinan
		UC17-TC03	Mengedit data dengan mengosongkan <i>field</i>	Sistem menampilkan pesan error validasi
		UC18-TC01	Menekan tombol hapus kepemimpinan	Data berhasil dihapus dan menampilkan semua data kepemimpinan
2.	Menu Visi Misi	UC01-TC01	Menekan menu visi misi	Menampilkan halaman visi misi
3.	Menu Struktur Organisasi	UC02-TC01	Menekan menu struktur organisasi	Menampilkan halaman struktur organisasi
4.	Menu Hymne	UC03-TC01	Menekan menu hymne	Menampilkan halaman hymne
		UC03-TC02	Menekan tombol putar video hymne	Video dapat diputar
5.	Menu Galeri Kegiatan	UC06-TC01	Menekan menu galeri kegiatan	Menampilkan halaman galeri kegiatan
		UC06-TC02	Menekan tombol lihat semua galeri kegiatan	Menampilkan halaman semua galeri kegiatan
		UC19-TC01	Menekan tombol tambah galeri kegiatan	Menampilkan form tambah galeri kegiatan
		UC19-TC02	Mengisi data galeri kegiatan dengan input valid	Data berhasil disimpan dan menampilkan semua data galeri kegiatan
		UC19-TC03	Mengisi form dengan data kosong	Sistem menampilkan pesan <i>error</i> validasi
		UC20-TC01	Menekan tombol <i>edit</i> galeri kegiatan	Menampilkan form <i>edit</i> galeri kegiatan
		UC20-TC02	Mengedit data dengan input valid	Data berhasil disimpan dan menampilkan semua data galeri kegiatan

		UC20-TC03	Mengedit data dengan mengosongkan <i>field</i>	Sistem menampilkan pesan <i>error</i> validasi
		UC21-TC01	Menekan tombol hapus galeri kegiatan	Data berhasil dihapus dan menampilkan semua data galeri kegiatan
6.	Menu Prestasi	UC05-TC01	Menekan menu Prestasi	Menampilkan halaman Prestasi
		UC05-TC02	Menekan tombol lihat semua Prestasi	Menampilkan halaman semua Prestasi
		UC10-TC01	Menekan tombol tambah Prestasi	Menampilkan form tambah Prestasi
		UC10-TC02	Mengisi data Prestasi dengan input valid	Data berhasil disimpan dan menampilkan semua data Prestasi
		UC10-TC03	Mengisi form dengan data kosong	Sistem menampilkan pesan <i>error</i> validasi
		UC11-TC01	Menekan tombol edit Prestasi	Menampilkan form edit Prestasi
		UC11-TC02	Mengedit data dengan input valid	Data berhasil disimpan dan menampilkan semua data Prestasi
		UC11-TC03	Mengedit data dengan mengosongkan <i>field</i>	Sistem menampilkan pesan error validasi
		UC12-TC01	Menekan tombol hapus Prestasi	Data berhasil dihapus dan menampilkan semua data Prestasi
7.	Menu Ekstrakurikuler	UC07-TC01	Menekan menu ekstrakurikuler	Menampilkan halaman ekstrakurikuler
		UC13-TC01	Menekan tombol tambah ekstrakurikuler	Menampilkan form tambah ekstrakurikuler
		UC13-TC02	Mengisi data ekstrakurikuler dengan input valid	Data berhasil disimpan dan menampilkan semua data ekstrakurikuler

		UC13-TC03	Mengisi form dengan data kosong	Sistem menampilkan pesan <i>error</i> validasi
		UC14-TC01	Menekan tombol <i>edit</i> ekstrakurikuler	Menampilkan form <i>edit</i> ekstrakurikuler
		UC14-TC02	Mengedit data dengan input valid	Data berhasil disimpan dan menampilkan semua data ekstrakurikuler
		UC14-TC03	Mengedit data dengan mengosongkan <i>field</i>	Sistem menampilkan pesan error validasi
		UC15-TC05	Menekan tombol hapus ekstrakurikuler	Data berhasil dihapus dan menampilkan semua data ekstrakurikuler
8.	Menu <i>Ranking</i>	UC08-TC01	Menekan menu <i>ranking</i>	Menampilkan halaman <i>ranking</i>
		UC08-TC02	Mengisi form filter berdasarkan kelas	Menampilkan halaman <i>ranking</i> berdasarkan kelas yang dicari
		UC08-TC03	Mengisi form filter berdasarkan nama	Menampilkan <i>ranking</i> dari nama yang dicari
9.	Menu <i>E-Raport</i>	UC09-TC01	Menekan menu <i>E-Raport</i>	Menampilkan data siswa pada halaman <i>E-Raport</i>
		UC09-TC02	Memilih nama salah satu siswa pada halaman <i>e-raport</i>	Menampilkan data <i>raport</i> siswa yang telah dipilih
		UC09-TC03	Menekan tombol cetak <i>e-raport</i>	Berhasil mencetak <i>e-raport</i>

Tabel 29. User Acceptance Testing Scenario

No.	Pernyataan	Skala
1.	Penggunaan warna tulisan dengan latar belakang sudah sesuai.	1-5

2.	Menu atau fitur yang tersedia di Sistem Informasi Akademik ini mudah dipahami.	1-5
3.	Waktu yang dibutuhkan untuk mengoperasikan sistem ini cukup efisien.	1-5
4.	Fitur-fitur dan tombol yang disediakan dalam sistem berfungsi dengan baik.	1-5
5.	Sistem ini membantu meningkatkan efektivitas kerja pengguna.	1-5
6.	Respon sistem saat digunakan terasa cepat dan stabil.	1-5
7.	Sistem memiliki tampilan yang menarik dan tidak membosankan.	1-5
8.	Sistem mampu berfungsi dengan baik di berbagai perangkat atau browser yang berbeda.	1-5

3.2.3. Tahap Penulisan Laporan

Tahap penulisan laporan bertujuan untuk menyajikan hasil penelitian secara lengkap dan mendokumentasikan seluruh proses yang telah dilakukan. Laporan ini diharapkan dapat memudahkan pembaca memahami langkah-langkah penelitian serta menjadi referensi untuk penelitian berikutnya.

V. KESIMPULAN DAN SARAN

5.1. Kesimpulan

1. Penelitian ini berhasil mengembangkan modul Informasi Publik pada Sistem Informasi Akademik berbasis website di SMPN 29 Bandar Lampung yang menyediakan akses informasi secara terpusat dan mudah diakses oleh masyarakat umum secara online. Sistem ini menjawab rumusan masalah pertama dan tujuan penelitian terkait penyediaan akses informasi yang cepat, mudah, dan transparan.
2. Modul Evaluasi yang dikembangkan memungkinkan siswa dan orang tua memantau hasil belajar secara real-time dan transparan melalui fitur *ranking* dan *e-raport*. Sistem ini mendukung peningkatan transparansi hasil akademik dan memberikan kontribusi dalam memotivasi semangat belajar siswa, dikarenakan data *ranking* yang ditampilkan selalu terupdate secara *realtime* setiap ada perubahan nilai, maka memungkinkan siswa untuk segera mengetahui jika peringkat mereka mengalami penurunan, sehingga terdorong untuk belajar lebih giat demi memperbaiki peringkat tersebut. Evaluasi efisiensi yang telah dilakukan juga menunjukkan sistem ini dapat mengatasi kesalahan kesalahan yang sering terjadi dalam proses manual. Sesuai dengan rumusan masalah kedua dan tujuan penelitian.

5.2. Saran

Berdasarkan implementasi sistem yang telah dikembangkan, terdapat beberapa saran untuk pengembangan lebih lanjut agar sistem dapat menjadi lebih baik:

1. Menambahkan fitur perpustakaan untuk memudahkan pengelolaan peminjaman dan pengembalian buku secara digital, sehingga pelayanan perpustakaan menjadi lebih efisien dan terintegrasi dengan sistem akademik.
2. Menyediakan fitur bahan ajar yang memungkinkan guru mengunggah materi pembelajaran secara online, guna mendukung akses siswa terhadap materi pembelajaran kapan saja dan di mana saja.

DAFTAR PUSTAKA

- Adi Saputra, B., Nabilah Kurnia Illahi, F., & Mukaromah, S. (2022). Audit Sistem Informasi Akademik Stikes Salsabila Menggunakan Cobit 5 Domain Dss. *Jurnal Ilmiah Ilmu Komputer*, 8(1), 09–13. <https://doi.org/10.35329/jiik.v8i1.209>
- Aliman, W. (2021). Perancangan Perangkat Lunak untuk Menggambar Diagram Berbasis Android. *Syntax Literate ; Jurnal Ilmiah Indonesia*, 6(6), 3091. <https://doi.org/10.36418/syntax-literate.v6i6.1404>
- Ardiansah, T., Rahmanto, Y., & Amir, Z. (2023). Penerapan Extreme Programming Dalam Sistem Informasi Akademik SDN Kuala Teladas. *Journal of Information Technology, Software Engineering and Computer Science (ITSECS)*, 1(2). <https://doi.org/10.58602/itsecs.v1i2.25>
- Ariansyah, P. M., & Wijaya, K. (2021). Rancang Bangun Sistem Informasi Akademik Berbasis Web: Studi Kasus: SD Negeri 18 Tanah Abang. *Jurnal Pengembangan Sistem Informasi Dan Informatika*, 2(3), 138–156. <https://doi.org/10.47747/jpsii.v2i3.562>
- Chamida, M. A., Susanto, A., & Latubessy, A. (2021). Analisa User Acceptance Testing Terhadap Sistem Informasi Pengelolaan Bedah Rumah Di Dinas Perumahan Rakyat Dan Kawasan Permukiman Kabupaten Jepara. *Indonesian Journal of Technology, Informatics and Science (IJTIS)*, 3(1), 36–41. <https://doi.org/10.24176/ijtis.v3i1.7531>
- Coyanda, J. R., Ariati, N., Sunardi, H., & K. Ghazali, K. G. (2022). Sistem Informasi Ekspedisi Barang Pada PT. New Power Global Energy Dengan Menggunakan

- Metode Extreme Programming. *Jurnal Ilmiah Informatika Global*, 13(2), 94–99. <https://doi.org/10.36982/jiig.v13i2.2295>
- Damanik, B. (2021). Rancangan Sistem Informasi Smp Negeri 1 Tuhemberua Kabupaten Nias Utara Menggunakan Php Codeigniter. *Jurnal Mahajana Informasi*, 6(1), 6–15. <https://doi.org/10.51544/jurnalmi.v6i1.1979>
- Dristyan, F., Syafnur, A., & Tinggi Manajemen Informatika dan Komputer Royal, S. (2021). SISTEM INFORMASI GEOGRAFIS PENYEBARAN COVID-19 DI KABUPATEN ASAHAN MENGGUNAKAN FRAMEWORK CODEIGNITER 4. In *Journal of Science and Social Research* (Issue 3). <http://jurnal.goretanpena.com/index.php/JSSR>
- Elyati, E., Idi, A., & Samiha, Y. T. (2022). Sekolah/Madrasah Sebagai Organisasi. *Jurnal Pendidikan Islam*, 173–190. <http://jurnal.staialhidayahbogor.ac.id/index.php/ei/article/view/3376%0Ahttp://jurnal.staialhidayahbogor.ac.id/index.php/ei/article/viewFile/3376/1355>
- Hakim, Z., & Meilina, P. (2022). *SISTEM INFORMASI AKADEMIK BERBASIS WEBISTE (STUDI KASUS: SMPIT AVICENNA)* (Vol. 12, Issue 3). <https://jurnal.umj.ac.id/index.php/just-it/index>
- Ikhsan, M., Helmina, Akbar, Z., Dani, R., & Ediansa, O. (2023). Sosialisasi dan Pelatihan Framework Codeigniter Untuk Mahasiswa Universitas Muhammadiyah Jambi. *ASPIRASI: Publikasi Hasil Pengabdian Dan Kegiatan Masyarakat*, 2(1), 70–76. <https://doi.org/10.61132/aspirasi.v2i1.138>
- Kustiawan, D., Cholifah, W. N., Destriana, R., & Heriyani, N. (2022). Rancang Bangun Sistem Informasi Akuntansi Pengelolaan Koperasi Menggunakan Metode Extreme Programming. *Jurnal Teknologi Dan Informasi*, 12(1), 78–92. <https://doi.org/10.34010/jati.v12i1.6756>
- Made, N., Febriyanti, D., KOMPIANG, A. A., Sudana, O., & Piarsa, N. (2021). *Implementasi Black Box Testing pada Sistem Informasi Manajemen Dosen* (Vol. 2, Issue 3).

- Ngantung, R. K., & Pakereng, M. A. I. (2021). Model Pengembangan Sistem Informasi Akademik Berbasis User Centered Design Menerapkan Framework Flask Python. *Jurnal Media Informatika Budidarma*, 5(3), 1052. <https://doi.org/10.30865/mib.v5i3.3054>
- Pratama, A. P., & Kamisutara, M. (2021). Pengembangan Sistem Informasi Akademik Berbasis Mobile Menggunakan Flutter Di Universitas Narotama Surabaya. *Network Engineering Research Operation*, 6(2), 145. <https://doi.org/10.21107/nero.v6i2.238>
- Putra, E., Losi, R. V., & Harahap, S. P. N. (2023). Sistem Aplikasi Kehadiran Staff Desa Berbasis Web Dengan Menggunakan Php Dan Mysql: Studi Desa Besilam Kecamatan Padang Tualang. *Jurnal Inovasi Pendidikan Dan Teknologi Informasi (JIPTI)*, 4(2), 201–213. <https://doi.org/10.52060/pti.v4i2.1472>
- Rakian, G. K. K., Mewengkang, A., & Palilingan, V. R. (2022). EduTIK: Jurnal Pendidikan Teknologi Informasi dan Komunikasi Volume 2 Nomor 1, Februari 2022. *EduTIK: Jurnal Pendidikan Teknologi Informasi Dan Komunikasi*, 2(April), 291.
- Ramdany, S. W., Aulia Kaidar, S., Aguchino, B., Amelia, C., Putri, A., & Anggie, R. (2024). Penerapan UML Class Diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web. In *Journal of Industrial and Engineering System* (Vol. 5, Issue 1).
- Rumbaugh, J., Jacobson, I., & Booch, G. (2021). The Unified Modeling Language Reference Manual. In *Journal of Chemical Information and Modeling* (Vol. 53, Issue 9).
- Sahat, P., Sihombing, M., Pradana, F., & Purnomo, W. (2023). *Pengembangan Sistem Informasi Akademik berbasis Web di SMP Negeri 3 Sidikalang* (Vol. 7, Issue 3). <http://j-ptiik.ub.ac.id>

- Septiani, N. A., & Habibie, F. Y. (2022). Penggunaan Metode Extreme Programming Pada Perancangan Sistem Informasi Pelayanan Publik. *Jurnal Sistem Komputer Dan Informatika (JSON)*, 3(3), 341. <https://doi.org/10.30865/json.v3i3.3931>
- Solahudin, M. (2021). Rancang Bangun Sistem Informasi Akademik Sekolah (SIAS) Berbasis Website. *DoubleClick: Journal of Computer and Information Technology*, 4(2), 107. <https://doi.org/10.25273/doubleclick.v4i2.8315>
- Suprpto, E. (2021). User Acceptance Testing (UAT) Refreshment PBX Outlet Site BNI Kanwil Padang. *Jurnal Civronlit Unbari*, 6(2), 54. <https://doi.org/10.33087/civronlit.v6i2.85>
- Sutedi, A., Septiana, Y., & Abdul Halim, R. (2021). Sistem Informasi Akademik Santri Berbasis Web di Pondok Pesantren. *Jurnal Algoritma*, 18(1), 151–161. <https://doi.org/10.33364/algoritma/v.18-1.934>