

TUGAS AKHIR

**RANCANG BANGUN APLIKASI PENJURUSAN
DI SMAN 1 KAPUAS HILIR MENGGUNAKAN METODE
SIMPLE MULTI ATTRIBUTE RATING TECHNIQUE (SMART) BERBASIS
WEB**



Disusun Oleh:

NATALIUS

NIM. DBC 114 059

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2019**

**RANCANG BANGUN APLIKASI PENJURUSAN
DI SMAN 1 KAPUAS HILIR MENGGUNAKAN METODE
SIMPLE MULTY ATTRIBUTE RATING (SMART) BERBASIS WEB**

TUGAS AKHIR

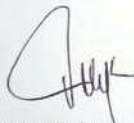
Sebagai salah satu syarat untuk menyelesaikan Program Strata-1
pada Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya

Oleh :

NATALIUS
NIM. DBC 114 059

Disetujui untuk diajukan dalam Sidang Tugas Akhir

Pembimbing I



DEDDY RONALDO, ST., MT
NIP. 19801226 200812 1 002

Pembimbing II



DEVI KAROLITA, S.KOM., M.KOM
NIP. 19821206 200604 2 001

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2019**

TUGAS AKHIR

**RANCANG BANGUN APLIKASI PENJURUSAN
DI SMAN 1 KAPUAS HILIR MENGGUNAKAN METODE
SIMPLE MULTY ATTRIBUTE RATING (SMART) BERBASIS WEB**

Sebagai salah satu syarat untuk menyelesaikan Program Strata-1 pada Jurusan Teknik
Informatika Fakultas Teknik Universitas Palangka Raya

Disusun Oleh :

NATALIUS
NIM. DBC 114 059

Telah dipertahankan di depan Tim Penguji, pada:

Hari/Tanggal : Kamis, 13 Juni 2019

Waktu : 12.00 – 14.00 WIB

- | | | |
|--|---------|-----------------|
| 1. Nova Noor Kamala Sari, ST., M.Kom
NIP. 19890407 201504 2 004 | : | (Ketua) |
| 2. Deddy Ronaldo, ST., MT
NIP. 19801226 200812 1 002 | : | (Pembimbing I) |
| 3. Devi Karolita, S.Kom., M.Kom
NIP. 19821206 200604 2 001 | : | (Pembimbing II) |
| 4. Ariesta Lestari, S.Kom., M.Cs
NIP. 19800322 200501 2 004 | : | (Anggota) |
| 5. Emy Dwi Oktaviani, ST., M.Kom
NIP. 19811003 200604 2 001 | : | (Anggota) |

Mengetahui :

Fakultas Teknik
Universitas Palangka Raya
Dekan,



Ir. WALUYO NUSWANTORO, MT
NIP. 19651119 199302 1 001

Jurusan Teknik Informatika
Fakultas Teknik Universitas Palangka Raya
Ketua Jurusan,



ABERTUN SAGIT SAHAY, ST., M.Eng
NIP. 19751212 200312 1 002

PERNYATAAN

Dengan ini saya menyatakan dengan sebenar-benarnya bahwa dalam Tugas Akhir ini tidak terdapat karya ilmiah yang pernah diajukan untuk memperoleh gelar kesarjanaan disuatu Perguruan Tinggi, serta tidak terdapat karya ilmiah atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam Tugas Akhir ini dan disebutkan dalam Tinjauan Pustaka.

Palangka Raya, Juni 2019



RIWAYAT PENYUSUN

DATA PRIBADI

Nama : NATALIUS
NIM : DBC 114 059
Tempat, Tanggal Lahir : Kuala Kapuas, 12 Desember 1995
Status : Belum Menikah
Agama : Kristen Protestan
Pekerjaan : Mahasiswa
Alamat : Jl. Hausman Baboe No. 53
Nama Ayah : RUBIADI
Nama Ibu : HELTHA



Riwayat Pendidikan *)

SD : SDN 2 Hampatung (Tahun Lulus 2007)
SMP : SMP Negeri 1 Kapuas Hilir (Tahun Lulus 2010)
SMA : SMA Negeri 1 Kapuas Hilir (Tahun Lulus 2013)

Palangka Raya, Juni 2019

NATALIUS
DBC 114 059

Keterangan:

*)Nama, Tempat, Tahun Lulus

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa karena atas kasih setia-Nya yang besar akhirnya saya dapat menyelesaikan Laporan Tugas Akhir yaitu “Rancang Bangun Aplikasi Penjurusan Di SMA Negeri 1 Kapuas Hilir Menggunakan Metode *Simple Multi Attribute Rating Technique (SMART)* Berbasis WEB”.

Saya selaku penulis menyadari dalam penyelesaian Tugas Akhir ini tidak lepas dari bimbingan, arahan dan bantuan dari berbagai pihak. Dalam kesempatan ini perkenankan saya untuk mengucapkan terima kasih kepada:

1. Tuhan Yesus Kristus yang selalu memberkati dan memberikan kesehatan serta memberikan talenta yang luar biasa agar dapat saya pakai untuk menjadi berkat untuk orang lain.
2. Bapak Deddy Ronaldo, ST., MT selaku Dosen Pembimbing I saya dan Ibu Devi Karolita, S.Kom., M.Kom selaku Dosen Pembimbing II saya yang telah meluangkan banyak waktu, tenaga dan pikiran di dalam memberikan pengarahan dalam penulisan Tugas Akhir ini.
3. Nova Noor Kamala Sari, S.T., M.Kom; Ariesta Lestari, S.Kom., M.Kom; dan Enny Dwi Oktaviyani, ST., M.Kom. selaku dosen penguji yang telah memberikan banyak masukan untuk tiap perbaikan didalam program maupun laporan.
4. Ayah dan Ibu tercinta serta kakak dan adik tersayang yang selalu memberikan dukungan dan semangat kepada saya dalam menyelesaikan Tugas Akhir ini.
5. Nadya Chitayae yang selalu memberikan dukungan, motivasi dan doa dengan setulus hati baik itu dari Pogram Profesional, Kerja Praktek dan sampai menyelesaikan Tugas Akhir ini.
6. Bang Meycelino A. Taguh, S.T yang selalu membimbing dan memberi arahan baik dalam perkuliahan maupun sampai pada Tugas akhir ini.
7. Semua dosen di Teknik Informatika Universitas Palangka Raya khususnya kepada Bapak V. Abdi Gunawan, S.T, M.T., Agus S. Saragih, S.T., M.Eng. dan Ade Chandra, S.Kom., M.Cs yang sudah memberikan ilmu dan pengetahuan yang amat banyak dalam perkuliahan dan dalam pelaksanaan tes SBMPTN yang telah mengikut sertakan saya sebagai teknisi mulai dari tahun 2017 sampai dengan 2019.
8. Teman-teman Teknik Informatika khususnya Angkatan 2014 serta teman-teman (Kehat, Aderay, Daniel dll yang tidak dapat saya sebutkan satu persatu) yang berjuang bersama-sama dan selalu memberikan hiburan dan semangat.
9. Semua pihak yang telah membantu dalam menyelesaikan Tugas Akhir ini baik secara langsung maupun tidak langsung.

Saya berharap program dan laporan tugas akhir ini dapat berguna dan bermanfaat bagi kita semua terkhusus bagi semua mahasiswa Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya. Atas perhatian dan partisipasinya, saya mengucapkan terima kasih.

Palangka Raya, Juni 2019

Penulis

**RANCANG BANGUN APLIKASI PENJURUSAN
DI SMAN 1 KAPUAS HILIR MENGGUNAKAN METODE
SIMPLE MULTI ATTRIBUTE RATING TECHNIQUE (SMART) BERBASIS WEB**

NATALIUS (DBC 114 059)

Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya
Kampus Tunjung Nyaho Jalan Yos Sudarso, Palangka Raya 73112
Email: natalius.it@mhs.it.upr.ac.id

ABSTRAK

Pada SMAN 1 Kapuas Hilir dalam proses belajar mengajar serta dalam peminatan jurusan telah berpacu pada kurikulum 2013. Diperlukan ketepatan dalam penentuan pemilihan jurusan karena akan berdampak pada masa depan siswa. Akan tetapi dalam penentuan jurusan siswa ini masih kurang efektif karena dalam penyeleksian penjurusan siswa hanya dengan menghitung nilai akademik sedangkan masih ada beberapa kriteria lain yang juga harus dipertimbangkan dalam penentuan jurusan. Pada proses penyeleksian penjurusan ini memiliki kelemahan karena membutuhkan waktu cukup lama terutama pada saat menghitung nilai akademik masing-masing siswa yang cukup banyak bisa terjadi banyak kekeliruan karena belum tersedianya sistem khusus untuk mendukung perhitungan tersebut. Kemudian pada proses pengambilan formulir penjurusan, siswa diharuskan datang ke sekolah untuk mendatangi wakasek bidang kesiswaan untuk mengambil formulir penjurusan, hal ini juga membutuhkan waktu cukup lama. Sehingga, penyelesaian dari masalah ini adalah dibangunnya sebuah Aplikasi Penjurusan Di SMAN 1 Kapuas Hilir menggunakan metode *Simple Multi Attribute Rating Technique* (SMART) berbasis *WEB* yang bertujuan untuk membantu menunjang keputusan pihak sekolah dalam melakukan penyeleksian dan penjurusan siswa tiap siswa.

Dalam mengembangkan aplikasi penjurusan siswa ini digunakan metodologi pengembangan perangkat lunak *Waterfall* menurut Roger S. Pressman (2010) dengan tahapan, *Communication* (Komunikasi), *Planning* (Perencanaan), *Modeling* (Permodelan), *Construction* (Konstruksi), *Deployment* (Penyerahan). Perangkat lunak yang digunakan dalam membangun *website* ini adalah Notepad++, SQLyog dan XAMPP.

Sistem ini dapat berjalan dengan baik dan dapat dikatakan layak untuk digunakan sebagaimana telah dilakukan pengujian *blackbox* pada sistem dan uji coba terhadap pengguna melalui kuesioner yang diproses menggunakan perhitungan skala likert yang diuji oleh user yaitu Kepala Sekolah, Wakil Kepala Sekolah bagian Kesiswaan dan Siswa.

Kata Kunci: Penjurusan Siswa, Metode SMART, *Waterfall*

**DESIGN AND DEVELOPMENT APPLICATION TO CONDUCT
THE SELECTION AND SCREENING OF STUDENT MAJORS
IN SMAN 1 KAPUAS HILIR USING THE METHOD SIMPLE MULTI ATTRIBUTE
RATING TECHNIQUE (SMART) BASED ON WEB**

NATALIUS (DBC 114 059)

*Department of Informatics Engineering, Faculty of Engineering
University of Palangka Raya
Tanjung Nyaho Campus St. Yos Sudarso Palangka Raya 73112
Email : natalius.it@mhs.it.upr.ac.id*

ABSTRACT

At SMAN 1 Kapuas Hilir in the teaching and learning process as well as majoring in majors have raced on the 2013 curriculum. Required accuracy in determining the selection of majors because it will have an impact on the future of students. However, in determining the student's majors it is still less effective because in the selection of student majors only by calculating academic values while there are still several other criteria that must also be considered in determining majors. In the process of selecting these majors has a weakness because it takes a long time, especially when calculating the academic value of each student which is quite a lot of mistakes can occur due to the unavailability of a special system to support these calculations. Then in the process of taking the majors form, students are required to come to school to visit the vice principal of the student field to take the majors form, this also takes quite a long time. WEB-based attribute Rating Technique (SMART) that aims to help support the school's decision to conduct departmental selection and screening of each student.

In developing this application, the methodology of Waterfall software development according to Roger S. Pressman (2010) with stages, communication, planning, modeling, construction and deployment. The software used in building this website is Notepad ++, SQLyog and XAMPP.

This system can run well and can be said to be feasible to use as has been done in the system blackbox testing and testing of users through questionnaires that are processed using a Likert scale calculation that is tested by the user namely Principal, Deputy Principal Student and Student section.

Keywords: *Student Majors, SMART method, Waterfall*

DAFTAR ISI

	Hal.
HALAMAN SAMPUL DEPAN	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN	iv
HALAMAN RIWAYAT PENYUSUN	v
KATA PENGANTAR	vi
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL	xv

BAB I PENDAHULUAN

1.1. Latar Belakang.....	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan Penelitian.....	2
1.5. Manfaat Penelitian.....	3
1.6. Metodologi Penelitian	3
1.7. Sistematika Penulisan.....	6
1.8. Jadwal Pelaksanaan.....	8
1.9. Tinjauan Pustaka.....	9

BAB II LANDASAN TEORI

2.1. Penjurusan Pada SMA Negeri 1 Kapuas Hilir.....	12
2.2. Metode <i>SMART</i> (<i>Simple Multi Attribute Rating Technique</i>).....	12
2.3. Pengertian <i>Website</i>	13
2.3.1. Jenis-Jenis <i>Website</i>	14

2.3.2. Peralatan Dalam Perancangan <i>Website</i>	15
2.4. Metode <i>Waterfall</i>	15
2.4.1. <i>Data Flow Diagram (DFD)</i>	17
2.4.2. <i>Entity Relationship Diagram (ERD)</i>	18
2.4.3 <i>Database</i>	21
2.4.4. <i>MySQL</i>	24
2.4.5. <i>XAMPP</i>	25
2.4.5.1. Asal Kata <i>XAMPP</i>	25
2.4.6. <i>Web Browser</i> dan <i>HTML</i>	26
2.4.7. <i>PHP</i>	26
2.4.7.1. Pengertian <i>PHP</i>	27
2.4.8. <i>Content Management System (CMS)</i>	28
2.4.9. <i>Cascading Style Sheet (CSS)</i>	29
2.4.10. <i>JQuery</i>	30
2.4.11. <i>Asynchronous Javascript and XML (AJAX)</i>	31
2.4.12. <i>Testing (Black Box)</i>	32
2.4.13. <i>Skala Likert</i>	33

BAB III ANALISIS DAN DESAIN

3.1. <i>Communication</i> (<i>Komunikasi</i>).....	34
3.1.1. <i>Teknologi</i>	34
3.1.1.1. <i>Hardware</i> / <i>Perangkat Keras</i>	34
3.1.1.2. <i>Software</i> / <i>Perangkat Lunak</i>	34
3.2. <i>Planning</i> (<i>Perencanaan</i>).....	35
3.2.1. <i>Analisis Sistem Lama</i>	35
3.2.2. <i>Analisis Bisnis Sistem Baru</i>	38
3.3. <i>Modeling</i> (<i>Permodelan</i>).....	44
3.3.1. <i>Analisis</i>	34
3.3.1.1. <i>Perhitungan Metode SMART (Simple Multi</i> <i>Attribute Rating Technique)</i>	44
3.3.1.2. <i>Data Flow Diagram (DFD)</i>	48

3.3.1.3. <i>Entity Relationship Diagram (ERD)</i>	63
3.3.2. Desain.....	65
3.3.2.1. Desain Tabel.....	65
3.3.2.2. Desain Navigasi.....	69
3.3.2.3. Desain Antarmuka Wakasek Bidang Kesiswaan.....	72
3.3.2.4. Desain Antarmuka Siswa.....	79
3.3.2.5. Desain Antarmuka Kepala Sekolah.....	82

BAB IV IMPLEMETASI DAN PENGUJIAN SISTEM

4.1. Implementasi.....	84
4.1.1. Implementasi Halaman Wakasek Bidang Kesiswaan.....	84
4.1.2. Implementasi Halaman Siswa.....	99
4.1.3. Implementasi Halaman Kepala Sekolah.....	102
4.2. Pengujian Sistem.....	104
4.2.1. Pengujian Kegiatan Yang Dilakukan Oleh Wakasek Bidang Kesiswaan.....	104
4.2.2. Pengujian Kegiatan Yang Dilakukan Oleh Siswa	116
4.2.3. Pengujian Kegiatan Yang Dilakukan Oleh Kepala Sekolah	120
4.3. Pengujian Sistem Terhadap Pengguna Menggunakan Skala Likert.....	104

BAB V PENUTUP

5.1 Kesimpulan	127
5.2 Saran	128

DAFTAR PUSTAKA	129
-----------------------------	-----

LAMPIRAN

DAFTAR GAMBAR

Gambar 1.1.	<i>Waterfall Modelling</i> (Roger S. Pressman, 2010).....	4
Gambar 2.1.	<i>Waterfall Modelling</i> (Roger S. Pressman, 2010).....	16
Gambar 2.2.	Contoh kardinalitas <i>one to one</i>	20
Gambar 2.3.	Contoh kardinalitas <i>one to many</i>	20
Gambar 2.4.	Contoh kardinalitas <i>many to many</i>	21
Gambar 3.1.	<i>Flowchart</i> Sistem Lama.....	37
Gambar 3.2.	<i>Flowchart</i> Sistem Baru.....	41
Gambar 3.3.	Diagram Konteks Aplikasi Penjurusan.....	49
Gambar 3.4.	DFD Level-1.....	52
Gambar 3.5.	DFD Level 2 Proses 2.0.....	57
Gambar 3.6.	DFD Level 2 Proses 3.0.....	57
Gambar 3.7.	DFD Level 2 Proses 5.0.....	58
Gambar 3.8.	DFD Level 2 Proses 6.0.....	58
Gambar 3.9.	DFD Level 2 Proses 11.1.....	59
Gambar 3.10.	DFD Level 2 Proses 13.0.....	59
Gambar 3.11.	DFD Level 3 Proses 2.2.....	60
Gambar 3.12.	DFD Level 3 Proses 2.3.....	60
Gambar 3.13.	DFD Level 3 Proses 3.1.....	61
Gambar 3.14.	DFD Level 3 Proses 3.2.....	61
Gambar 3.15.	DFD Level 3 Proses 3.3.....	62
Gambar 3.16.	DFD Level 4 Proses 3.2.2.....	62
Gambar 3.17.	DFD Level 4 Proses 3.3.2.....	62
Gambar 3.18.	ERD.....	64
Gambar 3.19.	Sitemap Wakasek Bidang Kesiswaan.....	70
Gambar 3.20.	Sitemap Siswa.....	71
Gambar 3.21.	Sitemap Kepala Sekolah.....	71
Gambar 3.22.	Halaman Login Wakasek Bidang Kesiswaan.....	72
Gambar 3.23.	Halaman Dashboard Wakasek Bidang Kesiswaan.....	72
Gambar 3.24.	Halaman Kelola Tahun Ajaran.....	73
Gambar 3.25.	Halaman Kelola Data Siswa.....	73

Gambar 3.26. Halaman Kelola Data Pimpinan.....	74
Gambar 3.27. Halaman Kelola Pembobotan Kriteria.....	74
Gambar 3.28. Halaman Kelola Himpunan Kriteria.....	75
Gambar 3.29. Halaman Kelola Klasifikasi.....	75
Gambar 3.30. Halaman Kelola Kuota.....	76
Gambar 3.31. Laporan Hasil Analisa.....	76
Gambar 3.32. Halaman Pengumuman Penjurusan.....	77
Gambar 3.33. Halaman Kelola Pengumuman Umum.....	77
Gambar 3.34. Halaman Waktu Pendaftaran.....	78
Gambar 3.35. Halaman Riwayat Penjurusan.....	78
Gambar 3.36. Halaman Kelola Akun Wakasek Bidang Kesiswaan.....	79
Gambar 3.37. Halaman Login Siswa.....	79
Gambar 3.38. Halaman Home.....	80
Gambar 3.39. Halaman Pendaftaran.....	80
Gambar 3.40. Halaman Pengumuman.....	81
Gambar 3.41. Halaman Panduan Pendaftaran.....	81
Gambar 3.42. Halaman Login Kepala Sekolah.....	82
Gambar 3.43. Halaman Home.....	82
Gambar 3.44. Halaman Validasi Hasil Seleksi.....	83
Gambar 4.1. Halaman Login Wakasek Bidang Kesiswaan.....	84
Gambar 4.2. Halaman Dashboard Wakasek Bidang Kesiswaan.....	85
Gambar 4.3. Halaman Kelola Tahun Ajaran	85
Gambar 4.4. Halaman Tambah Tahun Ajaran.....	86
Gambar 4.5. Halaman Ubah Tahun Ajaran.....	86
Gambar 4.6. Halaman Kelola Data Siswa.....	87
Gambar 4.7. Halaman Tambah Data Siswa.....	88
Gambar 4.8. Halaman Ubah Data Siswa.....	88
Gambar 4.9. Halaman Kelola Data Pimpinan.....	89
Gambar 4.10. Halaman Ubah Data Pimpinan	89
Gambar 4.11. Halaman Kelola Pembobotan Kriteria	90
Gambar 4.12. Halaman Ubah Pembobotan Kriteria	90

Gambar 4.13. Halaman Kelola Himpunan Kriteria	91
Gambar 4.14. Halaman Tambah Himpunan Kriteria	91
Gambar 4.15. Halaman Ubah Himpunan Kriteria	92
Gambar 4.16. Halaman Kelola Klasifikasi	92
Gambar 4.17. Halaman Lihat Data Klasifikasi	93
Gambar 4.18. Halaman Kelola Kuota	93
Gambar 4.19. Halaman Laporan Hasil Analisa	94
Gambar 4.20. Halaman Pengumuman Penjurusan	94
Gambar 4.21. Halaman Kelola Pengumuman Umum	95
Gambar 4.22. Halaman Tambah Pengumuman Umum	96
Gambar 4.23. Halaman Ubah Pengumuman Umum	96
Gambar 4.24. Halaman Kelola Waktu Pendaftaran	97
Gambar 4.25. Halaman Ubah Waktu Pendaftaran	97
Gambar 4.26. Halaman Riwayat Penjurusan	98
Gambar 4.27. Halaman Kelola Akun Wakasek Bidang Kesiswaan	98
Gambar 4.28. Halaman Login Siswa	99
Gambar 4.29. Login Berhasil	99
Gambar 4.30. Login Tidak Berhasil	99
Gambar 4.31. Halaman Home	100
Gambar 4.32. Halaman Form Pendaftaran.....	100
Gambar 4.33. Halaman Pengumuman.....	101
Gambar 4.34. Halaman Penduan Pendaftaran	101
Gambar 4.35. Halaman Kelola Akun Siswa	101
Gambar 4.36. Halaman Login Kepala Sekolah.....	102
Gambar 4.37. Login Berhasil	102
Gambar 4.38. Login Tidak Berhasil	102
Gambar 4.39. Halaman Home	103
Gambar 4.40. Halaman Kelola Hasil Seleksi	103
Gambar 4.41. Halaman Kelola Akun Kepala Sekolah	104

DAFTAR TABEL

Tabel 1.1.	Jadwal Pelaksanaan Tugas Akhir.....	8
Tabel 1.2.	Tinjauan Pustaka Tugas Akhir.....	9
Tabel 2.1.	Simbol-Symbol DFD.....	17
Tabel 2.2.	Tabel Komponen dan Simbol ERD.....	18
Tabel 3.1.	Rencana Kegiatan.....	43
Tabel 3.2.	Tabel Kriteria, Sub Kriteria dan Nilai Sub Kriteria.....	44
Tabel 3.3.	Tabel Normalisasi.....	45
Tabel 3.4.	Tabel Nilai Kriteria di Setiap Alternatif.....	46
Tabel 3.5.	Tabel menentukan nilai utility.....	46
Tabel 3.6.	Tabel Penentuan Nilai Total Masing-Masing Alternatif.....	47
Tabel 3.7.	Entitas.....	49
Tabel 3.8.	Definisi <i>Storage</i>	53
Tabel 3.9.	Definisi Proses.....	54
Tabel 3.10.	Tabel pengguna.....	65
Tabel 3.11.	Tabel siswa.....	66
Tabel 3.12.	Tabel himpunan kriteria.....	66
Tabel 3.13.	Tabel klasifikasi.....	67
Tabel 3.14.	Tabel kriteria.....	67
Tabel 3.15.	Tabel kuota.....	67
Tabel 3.16.	Tabel pengumuman.....	68
Tabel 3.17.	Tabel berkas.....	68
Tabel 3.18.	Tabel waktu pendaftaran.....	68
Tabel 3.19.	Tabel ranking.....	69
Tabel 3.20.	Tabel tahun ajaran.....	69
Tabel 4.1.	Blackbox Testing Proses Login Wakasek Bidang Kesiswaan	105
Tabel 4.2.	Blackbox Testing Halaman Utama Wakasek Bidang Kesiswaan	106
Tabel 4.3.	Blackbox Testing Halaman Kelola Data Tahun Ajaran	108
Tabel 4.4.	Blackbox Testing Halaman Kelola Data Siswa	108
Tabel 4.5.	Blackbox Testing Halaman Kelola Data Pimpinan	109
Tabel 4.6.	Blackbox Testing Halaman Kelola Pembobotan Kriteria	110

Tabel 4.7. Blackbox Testing Halaman Kelola Himpunan Kriteria	111
Tabel 4.8. Blackbox Testing Halaman Kelola Klasifikasi	112
Tabel 4.9. Blackbox Testing Halaman Kelola Kuota	113
Tabel 4.10. Blackbox Testing Halaman Laporan Hasil Analisa	113
Tabel 4.11. Blackbox Testing Halaman Pengumuman Penjurusan	114
Tabel 4.12. Blackbox Testing Halaman Pengumuman Umum	114
Tabel 4.13. Blackbox Testing Halaman Kelola Waktu Pendaftaran	115
Tabel 4.14. Blackbox Testing Halaman Kelola Akun Wakasek Bidang Kesiswaan	116
Tabel 4.15. Blackbox Testing Proses Login Siswa	116
Tabel 4.16. Blackbox Testing Halaman Utama Siswa	117
Tabel 4.17. Blackbox Testing Halaman Pendaftaran Penjurusan	118
Tabel 4.18. Blackbox Testing Halaman Kelola Akun Siswa	119
Tabel 4.19. Blackbox Testing Proses Login Kepala Sekolah	119
Tabel 4.20. Blackbox Testing Halaman Utama Kepala Sekolah	120
Tabel 4.21. Blackbox Testing Halaman Kelola Hasil Seleksi	121
Tabel 4.22. Blackbox Testing Halaman Kelola Akun Kepala Sekolah	121
Tabel 4.23. Skala Jawaban.....	122
Tabel 4.24. Skor Ideal.....	122
Tabel 4.25. Kuesioner Pengguna Pernyataan 1.....	123
Tabel 4.26. Kuesioner Pengguna Pernyataan 2.....	124
Tabel 4.27. Kuesioner Pengguna Pernyataan 3.....	124
Tabel 4.28. Kuesioner Pengguna Pernyataan 4.....	125
Tabel 4.29. Kuesioner Pengguna Pernyataan 5.....	125
Tabel 4.30. Hasil Akhir Kuesioner.....	126

BAB I

PENDAHULUAN

1.1. LATAR BELAKANG

Pada SMAN 1 Kapuas Hilir dalam proses belajar mengajar serta dalam peminatan jurusan telah berpacu pada kurikulum 2013. Peminatan jurusan di kurikulum 2013 tidak lagi dilakukan pada kelas XI, melainkan mulai dari kelas X setelah siswa dinyatakan diterima di sekolah tersebut. Jika pada kurikulum sebelumnya terdapat jurusan Ilmu Pengetahuan Alam (IPA) dan Ilmu Pengetahuan Sosial (IPS), maka pada kurikulum 2013 ini peminatan jurusannya adalah Matematika dan Ilmu Alam (MIA) serta Ilmu-Ilmu Sosial (IIS).

Setelah siswa dinyatakan diterima di SMAN 1 Kapuas Hilir, siswa akan diminta untuk melengkapi formulir peminatan jurusan serta mengumpulkan masing-masing nilai mereka. Namun penentuan siswa yang masuk ke jurusan MIA dan IIS ini tetap akan diseleksi dan ditentukan sekolah, yaitu oleh Wakil Kepala Sekolah Bagian Kesiswaan, hal ini dilakukan karena terdapat syarat-syarat serta terbatasnya kuota siswa yang masuk ke kelas MIA atau IIS.

Diperlukan ketepatan dalam penentuan pemilihan jurusan karena akan berdampak pada masa depan siswa. Akan tetapi dalam penentuan jurusan siswa ini masih kurang efektif karena dalam penyeleksian penjurusan siswa hanya dengan menghitung nilai akademik sedangkan masih ada beberapa kriteria lain yang juga harus dipertimbangkan dalam penentuan jurusan. Pada proses penyeleksian penjurusan ini memiliki kelemahan karena membutuhkan waktu cukup lama terutama pada saat menghitung nilai akademik masing-masing siswa yang cukup banyak bisa terjadi banyak kekeliruan karena belum tersedianya sistem khusus untuk mendukung perhitungan tersebut. Kemudian pada proses pengambilan formulir penjurusan, siswa diharuskan datang ke sekolah untuk mendatangi wakasek bidang kesiswaan untuk mengambil formulir penjurusan, hal ini juga membutuhkan waktu cukup lama.

Berdasarkan permasalahan di atas, maka akan dibuat sistem aplikasi penjurusan dengan metode *SMART (Simple Multi Attribute Rating Technique)* berbasis *WEB* yang merupakan teknik pembuatan keputusan multiatribut untuk

memilih beberapa alternatif yang memiliki sekumpulan kriteria, nilai dan bobot yang menggambarkan seberapa penting kriteria tersebut sehingga akan diperoleh alternatif terbaik. yang diharapkan dapat membantu SMAN 1 Kapuas Hilir dalam menentukan jurusan tiap siswa.

1.2. RUMUSAN MASALAH

Berdasarkan latar belakang yang telah diuraikan di atas, maka rumusan masalahnya yaitu bagaimana merancang dan membangun aplikasi penjurusan di SMAN 1 Kapuas Hilir menggunakan metode *Simple Multi Attribute Rating Technique* (SMART) berbasis *WEB* ?

1.3. BATASAN MASALAH

Batasan masalah dalam pembuatan program ini dilakukan agar perancangan dan pembuatan *website* sistem penunjang keputusan seleksi pemilihan jurusan ini lebih teratur dan terarah. Berdasarkan rumusan masalah yang ada, batasan masalah yang akan dijadikan dasar desain *website* yang dibuat yaitu sebagai berikut.

1. Rancang Bangun Aplikasi Penjurusan Di SMAN 1 Kapuas Hilir Menggunakan Metode *Simple Multi Attribute Rating Technique* (SMART) Berbasis *WEB* dibuat menggunakan bahasa pemrograman *PHP* & database *MySQL*.
2. Aplikasi penjurusan berbasis *web ini* menggunakan 5 kriteria dan bobot masing-masing yang telah didiskusikan dengan pihak SMA Negeri 1 Kapuas Hilir untuk penentuan penjurusan, yaitu Nilai rata-rata akademik kelas VII – IX dengan bobot 30, nilai rata-rata IPA kelas VII – IX dengan bobot 25, nilai rata-rata IPS kelas VII – IX dengan bobot 25, prestasi dibidang IPA dengan bobot 10 dan minat siswa dengan bobot 10.

1.4. TUJUAN PENELITIAN

Tujuan dibuatnya aplikasi penjurusan dengan metode *SMART* (*Simple Multi Attribute Rating Technique*) berbasis *WEB* ini adalah sebagai berikut.

1. Membuat suatu aplikasi berbasis *WEB* yang dapat digunakan untuk menentukan jurusan tiap siswa.
2. Merancang aplikasi penjurusan siswa berbasis *web* dengan menggunakan metode *Simple Multi Attribute Rating Technique* (SMART) untuk menghasilkan hasil seleksi yang tepat berdasarkan penilaian kriteria-kriteria yang telah ditentukan.

1.5. MANFAAT PENELITIAN

Pembuatan aplikasi penjurusan dengan metode *SMART* (*Simple Multi Attribute Rating Technique*) berbasis *WEB* ini diharapkan memberi manfaat khususnya Wakil Kepala Sekolah Bagian Kesiswaan dalam proses penjurusan tiap siswa sesuai dengan minat dan kemampuannya berdasarkan kriteria yang ditentukan.

1.6. METODOLOGI PENELITIAN

Metode penelitian untuk membuat aplikasi penjurusan dengan metode *SMART* (*Simple Multi Attribute Rating Technique*) berbasis *WEB*, yaitu meliputi:

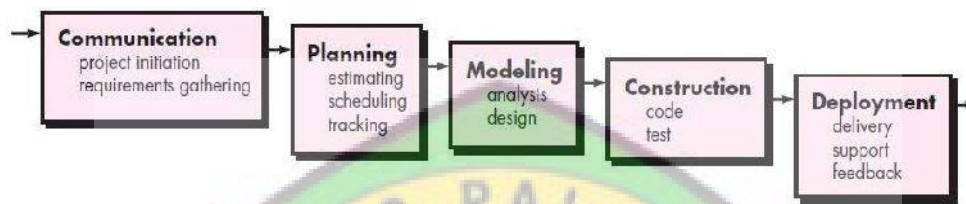
1. Metode Pengumpulan Data
Metode pengumpulan data merupakan tahap pengambilan data atau sampel yang berhubungan dengan permasalahan yang sedang dibahas. Pengumpulan data yang dilakukan pada proses ini, meliputi:
 - a. Observasi dengan mengamati dan mempelajari secara langsung proses penjurusan di SMAN 1 Kapuas Hilir.
 - b. Studi lapangan, yaitu melakukan wawancara dengan Wakil Kepala Sekolah Bagian Kesiswaan dengan meminta beberapa keterangan informasi mengenai penjurusan di SMAN 1 Kapuas Hilir.
 - c. Studi kepustakaan dan studi literatur, tahap ini dilakukan dengan cara mencari referensi seperti mempelajari teori-teori literatur dari buku dan informasi dari internet yang berhubungan dengan proses pembuatan aplikasi penjurusan berbasis *WEB*.

2. Metode Konsultasi

Mengkonsultasikan mengenai permasalahan yang akan dibahas kepada dosen pembimbing atau orang lain yang mengerti tentang objek program yang akan dibuat.

3. Metode Pengembangan Sistem

Metode yang digunakan dalam pengembangan sistem adalah model *Waterfall* (Roger S. Pressman, 2010) dengan tahap-tahap sebagai berikut:



Gambar1.1 Waterfall Modelling

(sumber : Roger S. Pressman, 2010)

a. *Communication* (Komunikasi)

Pada tahap ini merupakan tahap analisis terhadap kebutuhan software, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan Wakil Kepala Sekolah Bagian Kesiswaan, maupun mengumpulkan data-data yang diperlukan. Hasil pada tahap komunikasi yaitu data-data yang dibutuhkan seperti data siswa kelas X, data raport siswa kelas VII sampai dengan kelas IX dan buku pedoman yang berkaitan dengan penjurusan dalam pembuatan *website* dan teknologi yang akan digunakan.

b. *Planning* (Perencanaan)

Proses *planning* merupakan lanjutan dari proses *communication* (komunikasi). Tahapan ini akan menghasilkan data yang berhubungan dengan keinginan user dalam pembuatan software, termasuk rencana yang akan dilakukan.

c. *Modeling* (Permodelan)

Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding*. Proses ini terbagi menjadi 2, yaitu analisis dan desain. Pada analisis menggunakan *Data Flow Diagram (DFD)* serta *Entity Relationship Diagram*. Namun sebelum memasuki tahapan ini, user terlebih dahulu menganalisis metode yang digunakan untuk penjurusan yang dilakukan di SMAN 1 Kapus Hilir secara manual pada microsoft excel menggunakan metode *Simple Multi Attribute Rating Technique (SMART)*.

Pada desain akan mendesain tabel, desain navigasi menggunakan *Sitemap* dan desain antarmuka/representasi *interface*. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*.

d. *Construction* (Konstruksi)

Construction merupakan proses membuat kode. *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. *Programmer* akan menerjemahkan kebutuhan *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan aplikasi, artinya penggunaan computer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem dan uji coba terhadap pengguna melalui kuesioner. Tujuan *testing* dan uji coba terhadap pengguna adalah untuk menemukan kesalahan-kesalahan terhadap sistem dan mengetahui sistem yang dibuat layak digunakan atau tidak agar kemudian bisa diperbaiki. Untuk menterjemahkan kode-kode program menggunakan bahasa pemrograman PHP & MySQL dan pengujian dilakukan menggunakan Blackbox Testing serta uji coba terhadap pengguna dengan menyebarkan kuesioner yang kemudian diproses untuk dihitung menggunakan skala likert.

e. *Deployment* (Penyerahan)

Tahapan ini merupakan tahap akhir *final* dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan

pengkodean maka system yang sudah jadi akan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus evaluasi jika ada kekurangan dan dilakukan pemeliharaan secara berkala.

Akan tetapi, pada pembuatan aplikasi ini penulis hanya melakukan sampai pada tahapan *testing* (pengujian).

1.7. Sistematika Penulisan

Penulisan laporan Tugas Akhir ini disusun dalam lima bab, dengan menggunakan sistematika sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini diuraikan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, serta metodologi dan sistematika penulisan.

BAB II LANDASAN TEORI

Landasan teori memuat masalah tentang konsep dan prinsip dasar yang diperlukan untuk memecahkan masalah pekerjaan untuk merumuskan hipotesis. Landasan teori ini berbentuk permasalahan-permasalahan yang langsung berkaitan dengan permasalahan yang dikerjakan.

BAB III ANALISIS DAN DESAIN SISTEM

Pada bab ini berisi uraian tentang perancangan proses, perancangan basis data, perancangan antar muka yang meliputi perancangan *input* dan *output*, kebutuhan sistem, serta uraian tahapan-tahapan dari implementasi sistem yang digunakan yaitu rancangan penerapan, dan tahap penggunaan program.

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini berisi tentang penjelasan hasil implementasi dari *website* yang telah di buat, pengujian (*testing*) program dan uji coba terhadap pengguna dengan menyebarkan kuesioner. Pada tahap pengujian aplikasi penjurusan di SMAN 1 Kapuas Hilir menggunakan metode *Simple Multi Attribute Rating* (SMART) berbasis WEB ini akan di

gunakan *black box testing* dan tahap untuk menghitung hasil kuesioner menggunakan perhitungan skala likert.

BAB V PENUTUP

Bab ini terdiri dari kesimpulan dan saran. Kesimpulan yang berisi pernyataan singkat dan tepat yang dijabarkan dari hasil studi literatur atau landasan teori dan penyusunan laporan Tugas Akhir, sedangkan saran berupa perbaikan/peningkatan yang diperlukan saat ini ataupun pada masa yang akan datang yang berhubungan dengan pelaksanaan pembuatan Tugas Akhir.



1.8. JADWAL PELAKSANAAN

Adapun jadwal pelaksanaan dalam pembuatan program pada Tugas Akhir ini adalah sebagai berikut.

Tabel 1.1. Jadwal Pelaksanaan Tugas Akhir

No.	Kegiatan	Bulan																							
		Bulan ke 1				Bulan ke 2				Bulan ke 3				Bulan ke 4				Bulan ke 5				Bulan ke 6			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Penyusunan Proposal	■	■	■	■																				
2	Pengumpulan Proposal					■	■	■	■																
3	Seminar Proposal									■	■	■	■												
4	Communication(Komunikasi)																								
5	Planning(Perencanaan)																								
6	Modeling (Permodelan)													■	■	■	■								
7	Construction(Konstruksi)																	■	■	■	■	■	■	■	■
8	Seminar Tugas Akhir																					■			

1.9. TINJAUAN PUSTAKA

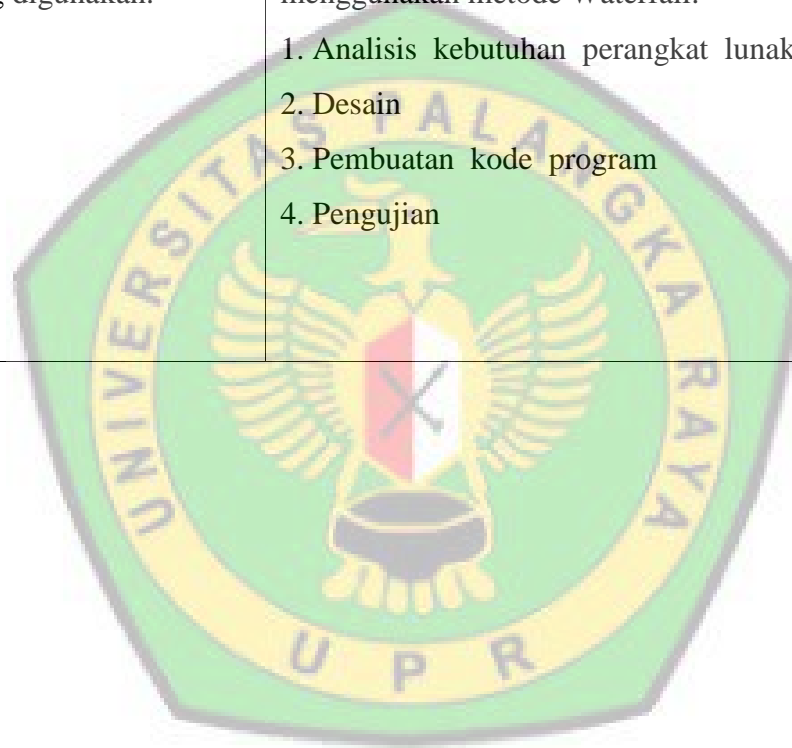
Dalam pengembangan penelitian yang berjudul “Rancang Bangun Aplikasi Penjurusan di SMAN 1 Kapuas Hilir Menggunakan Metode *Simple Multi Attribute Rating* (SMART) Berbasis *WEB*” ini akan dilakukan analisis sebagai pembandingan dengan aplikasi yang telah ada sebelumnya seperti pada Tabel 1.1.

Tabel 1.2. Tinjauan Pustaka Tugas Akhir

	I	II	III
Judul	Sistem Pendukung Keputusan Pemilihan Jurusan dengan Menggunakan Metode SAW (Simple Additive Weighting) di SMA 6 Tasikmalaya	Sistem Pendukung Keputusan Menentukan Kelayakan Bisnis Menerapkan <i>Simple Multi Attribute Rating Technique</i>(SMART)	Rancang Bangun Aplikasi Penjurusan di SMAN 1 Kapuas Hilir Menggunakan Metode Simple Multi Attribute Rating (SMART) Berbasis <i>WEB</i>
(Nama, Universitas, Tahun)	(Teuku Mufizar, Dede SyahrulAnwar, Epa Aprianis, STMIK Tasikmalaya, 2016)	(Diana, Universitas Bina Darma, 2016)	(Natalius, Universitas Palangkaraya, 2019)
Pengguna	Guru BK	Administrator	Wakil Kepala Sekolah Bagian Kesiswaan
Tujuan Aplikasi	Perancangan aplikasi ini di bangun bertujuan untuk membantu guru BK dalam menentukan penjurusan tiap siswa.	Penelitian ini bertujuan untuk membangun sistem pendukung keputusan yang dapat menentukan kelayakan bisnis sehingga dapat mengetahui usaha mana	a. Membuat suatu aplikasi berbasis <i>WEB</i> yang dapat digunakan untuk menentukan jurusan tiap siswa. b. Merancang aplikasi penjurusan

		<p>saja yang pantas dilanjutkan atau tidak dilanjutkan.</p>	<p>siswa berbasis <i>web</i> dengan menggunakan metode <i>Simple Multi Attribute Rating (SMART)</i> untuk menghasilkan hasil seleksi yang tepat berdasarkan penilaian kriteria-kriteria yang telah ditentukan.</p> <p>c. Menyelesaikan tugas sebagai syarat yang ditempuh pada mata kuliah Tugas Akhir (TA) di Jurusan Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya tahun 2019.</p>
Teknologi	<p>Program ini dibuat dalam aplikasi desktop dengan menggunakan bahasa pemrograman <i>java</i>.</p>	<p>Pada jurnal tidak disebutkan teknologi yang digunakan. Tetapi program ini diketahui berbasis desktop.</p>	<p>Teknologi yang digunakan dalam aplikasi ini adalah tool <i>xampp</i> sebagai <i>webserver (local host)</i>, bahasa pemrograman PHP dan HTML, MySQL untuk membuat <i>database</i> pada <i>website</i>, <i>chrome</i> sebagai <i>web browser</i>, <i>Notepad++</i> sebagai editor <i>coding</i>, <i>Adobe Photoshop</i> untuk</p>

			membuat desain grafis <i>website dan Edraw Max</i> untuk visualisasi.
Metodologi	Pada jurnal tidak disebutkan metodologi yang digunakan.	Metodologi Pengembangan Sistem menggunakan metode Waterfall: <ol style="list-style-type: none"> 1. Analisis kebutuhan perangkat lunak 2. Desain 3. Pembuatan kode program 4. Pengujian 	Metodologi Pengembangan Sistem menggunakan metode Waterfall menurut Roger S. Pressman, 2010: <ol style="list-style-type: none"> 1. <i>Communication</i>(Komunikasi) 2. <i>Planning</i>(Perencanaan) 3. <i>Modeling</i> (Permodelan) 4. <i>Construction</i>(Konstruksi) 5. <i>Deployment</i> (Penyerahan)



BAB II

LANDASAN TEORI

2.1 Penjurusan Pada SMA Negeri 1 Kapuas Hilir

Pada SMAN 1 Kapuas Hilir dalam proses belajar mengajar serta dalam peminatan jurusan telah berpacu pada kurikulum 2013. Peminatan jurusan di kurikulum 2013 tidak lagi dilakukan pada kelas XI, melainkan mulai dari kelas X setelah siswa dinyatakan diterima di sekolah tersebut. Jika pada kurikulum sebelumnya terdapat jurusan Ilmu Pengetahuan Alam (IPA) dan Ilmu Pengetahuan Sosial (IPS), maka pada kurikulum 2013 ini peminatan jurusannya adalah Matematika dan Ilmu Alam (MIA) serta Ilmu-Ilmu Sosial (IIS).

Setelah siswa dinyatakan diterima di SMAN 1 Kapuas Hilir, siswa akan diminta untuk melengkapi formulir peminatan jurusan serta mengumpulkan masing-masing nilai mereka. Namun penentuan siswa yang masuk ke jurusan MIA dan IIS ini tetap akan diseleksi dan ditentukan sekolah, yaitu oleh Wakil Kepala Sekolah Bagian Kesiswaan, hal ini dilakukan karena terdapat syarat-syarat serta terbatasnya kuota siswa yang masuk ke kelas MIA atau IIS.

2.2 Metode SMART (*Simple Multi Attribute Rating Technique*)

Simple Multi Attribute Rating Technique (SMART) merupakan teknik atau metode yang multi attribute dalam sistem pengambilan keputusan. Metode ini dikembangkan pada tahun 1977 oleh Edward. Decision maker harus memilih objek-objek yang berbeda dan memiliki kesempatan yang sama, disesuaikan dengan tujuan yang telah dirumuskan. Setiap objek-objek ini memiliki karakter, komponen, atau kriteria dalam keputusan. Namun dalam suatu kondisi tidak menutup kemungkinan adanya sub kriteria yang berhubungan dengan kriteria yang telah ada. Karakter, komponen atau kriteria keputusan ini memiliki nilai-nilai. Nilai ini di rata-rata dengan skala tertentu. Setiap kriteria memiliki bobot yang mendefinisikan seberapa penting kriteria tersebut dari kriteria yang lain. Pembobotan ini dilakukan untuk menilai setiap objek-objek yang berbeda yang memiliki kesempatan sama, sehingga mendapatkan objek terbaik. Pembobotan

pada SMART menggunakan skala antara 0 dan 1, sehingga mempermudah perhitungan dan perbandingan nilai pada masing-masing alternatif.

Adapun langkah-langkah menggunakan metode SMART (Edward, 1977):

Langkah 1 : Menentukan jumlah kriteria.

Langkah 2 : Sistem secara default memberikan skala 0 – 100 berdasarkan prioritas yang telah diinputkan kemudian dilakukan normalisasi.

$$\text{Rumus Normalisasi adalah : } \frac{W_j}{\sum W_j}$$

Keterangan :

W_j : bobot suatu kriteria

Langkah 3 : Memberikan nilai kriteria untuk setiap alternatif.

Langkah 4 : Menghitung nilai utility untuk setiap kriteria masing-masing.

$$u_i(a_i) = 100 \frac{(C_{out\ i} - C_{min})}{(C_{max} - C_{min})} \%$$

Keterangan :

$u_i(a_i)$: nilai utility kriteria ke – 1 untuk kriteria ke – i

C_{max} : nilai kriteria maxsimal

C_{min} : nilai kriteria minimal

$C_{out\ i}$: nilai kriteria ke – i

Langkah 5 : Menghitung nilai akhir masing-masing.

$$u(a_i) = \sum_{j=1}^m w_j u_i(a_i)$$

Keterangan :

$u(a_i)$: nilai total alternatif

w_j : hasil dari normalisasi bobot kriteria

$u_i(a_i)$: hasil penentuan nilai utility

2.3 Pengertian Website

Yuhefizar, dkk (2011:2) menyatakan *website* adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah *domain* yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Hubungan antara satu halaman *web* dengan *web* lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut

hypertext.

Domain adalah nama unik yang dimiliki oleh sebuah institusi sehingga biasanya diakses melalui *internet*, misalnya *lintau.com*, *yahoo.com*, *goggle.com* dan lain-lain. Untuk mendapat sebuah *domain* harus melakukan register pada register-register yang ditentukan. Istilah lain yang sering ditemui sehubungan dengan *website* adalah *homepage*. *Homepage* adalah halaman awal sebuah *domain*. Misalnya, membuka *website www.lintau.com*, halaman pertama yang muncul disebut *homepage* jika mengklik menu-menu yang ada dan meloncat ke lokasi yang lainnya disebut *web page* sedangkan keseluruhan isi/konten *domain* disebut *website*.

2.3.1 Jenis-Jenis Website

Menurut Yuhefizar, dkk (2011:3) seiring dengan perkembangan teknologi informasi yang begitu cepat, *website* juga mengalami perkembangan yang sangat berarti. Dalam pengelompokan jenis *web*, lebih diarahkan berdasarkan pada fungsi, fungsi dan bahasa pemrograman yang digunakan. Jenis-jenis *web* berdasarkan sifatnya adalah:

- a. *Website* dinamis merupakan sebuah *website* yang menyediakan konten/isi yang selalu berubah-ubah setiap saat. Misalnya *website* berita, seperti: *www.kompas.com*, *www.detik.com*, *www.polinpdg.ac.id* dan lain-lain.
- b. *Website* statis merupakan *website* yang kontennya sangat jarang diubah. Misalnya: *web* profil organisasi dan lain-lain.

Berdasarkan tujuannya, *website* dibagi atas :

1. *Personal web*, *website* yang berisi informasi pribadi seseorang.
2. *Corporate web*, *website* yang dimiliki oleh sebuah perusahaan
3. *Portal web*, *website* yang mempunyai banyak layanan, mulai dari layanan berita, *email* dan jasa-jasa lainnya.
4. *Forum web*, sebuah *web* yang bertujuan sebagai media diskusi

Disamping itu juga ada *website e-Governman*, *e-banking*, *e-payment*, *e-procurement* dan sebagainya. Ditinjau dari segi bahasa pemrograman yang digunakan, *website* terbagi atas :

1. *Server Side* merupakan *website* yang menggunakan bahasa pemrograman yang tergantung kepada tersedianya *server*. Seperti *PHP*, *ASP* dan lain sebagainya. Jika tidak ada *server*, *website* yang dibangun menggunakan bahasa pemrograman di atas tidak dapat berfungsi sebagaimana mestinya.
2. *Client Side*, adalah *website* yang tidak membutuhkan *server* dalam menjalankannya, cukup diakses melalui *browser* saja. Misalnya: *html*.

2.3.2 Peralatan Dalam Perancangan *Website*

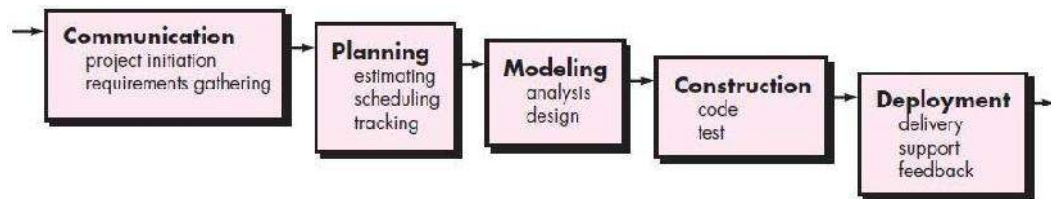
Menurut Yuhefizar, dkk (2011:5) dalam merancang dan membangun sebuah *website* yang baik diperlukan beberapa *skill*. Oleh karena itu biasanya dalam pengerjaan sebuah *website* dibutuhkan tim, agar tujuan pembuatan *web* yang baik tercapai. Berikut ini beberapa *skill* yang sebaiknya dimiliki oleh seorang *webmaster*, sebagai berikut :

1. *Skill* dalam bidang bahasa pemrograman. Misalnya: bahasa pemrograman *PHP*, *ASP*, *JSP*, *Javascript*, *HTML* dan lain-lain.
2. *Skill* dalam bidang desain. Misalnya: menguasai *software Photoshop*, *Illustrator*, *Paintshop Pro*, dan lain-lain.
3. *Skill* dalam bidang animasi. Misalnya: menguasai *Flash*, *Swish*, *Image Eeady*, *Swift*, *Gif Animator*, dan lain-lain.

2.4 Metode *Waterfall*

Metode yang digunakan dalam mengembangkan sistem informasi menggunakan Metode pengembangan perangkat lunak *Waterfall*. Menurut Roger S. Pressman (2010:46) metode ini terdapat 5 (lima) tahap untuk mengembangkan suatu perangkat lunak yaitu *Communication*, *Planning*, *Modeling*, *Construction*, *Deployment*. Dimana konsep dari metode ini adalah melihat suatu masalah secara sistematis dan terstruktur dari atas ke bawah.

Tahap-tahap pengembangan perangkat lunak metode *Waterfall* dapat dilihat pada gambar 2.1:



Gambar 2.1 Waterfall Modelling

(sumber : Roger S. Pressman, 2010)

Berikut ini akan diuraikan tahap-tahap pengembangan perangkat lunak dengan menggunakan metode *Waterfall*, yaitu :

Tahap 1 : Communication (Komunikasi)

Pada tahap ini merupakan tahap analisis terhadap kebutuhan software, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan Wakil Kepala Sekolah Bagian Kesiswaan, maupun mengumpulkan data-data yang diperlukan. Hasil pada tahap komunikasi yaitu data-data yang dibutuhkan seperti data siswa kelas X, data raport siswa kelas VII sampai dengan kelas IX dan buku pedoman yang berkaitan dengan penjurusan dalam pembuatan website dan teknologi yang akan digunakan.

Tahap 2 : Planning (Perencanaan)

Proses *planning* merupakan lanjutan dari proses *communication* (komunikasi). Tahapan ini akan menghasilkan data yang berhubungan dengan keinginan user dalam pembuatan software, termasuk rencana yang akan dilakukan.

Tahap 3 : Modeling (Permodelan)

Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan software yang dapat diperkirakan sebelum dibuat coding. Proses ini terbagi menjadi 2, yaitu analisis dan desain. Pada analisis menggunakan Data Flow Diagram (DFD) serta Entity Relationship Diagram. Namun sebelum memasuki tahapan ini, user terlebih dahulu menganalisis metode yang digunakan untuk penjurusan yang dilakukan di SMAN 1 Kapus Hilir secara manual pada microsoft excel menggunakan metode Simple Multi Attribute Rating Technique (SMART).

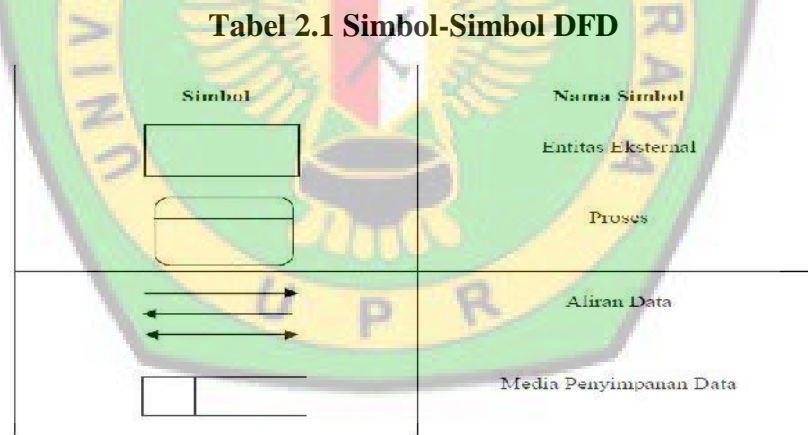
Pada desain akan mendesain tabel, desain navigasi menggunakan Sitemap dan desain antarmuka/representasi interface. Tahapan ini akan menghasilkan dokumen yang disebut software requirement.

2.4.1 Data Flow Diagram (DFD)

Analisis perancangan aplikasi penjurusan didefinisikan dengan menggunakan DFD (Data Flow Diagram). Semua aliran data yang masuk dan keluar melalui sistem digambarkan dengan DFD, sehingga dapat dilihat dengan jelas aliran data apa saja yang terlibat dalam proses di sistem ini.

Saputra dan Agustin (2012:28) Data Flow Diagram atau yang disingkat DFD merupakan diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas. DFD juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dan transformasi yang digunakan sebagai perjalanan data dari input atau masukan menuju keluaran atau output.

DFD menurut Gane dan Sarson, digambarkan dengan menggunakan empat simbol seperti pada Tabel 2.1.



Rosa dan Shalahuddin (2011:118) menyatakan tahapan-tahapan perancangan dengan menggunakan DFD seperti berikut :

1. Membuat DFD Level 0 atau sering disebut juga Context Diagram

DFD level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD level 1 merupakan hasil breakdown DFD level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD level 1 dapat di-breakdown menjadi DFD level 2. Modul mana saja yang harus di-breakdown lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-breakdown lagi. Untuk sebuah sistem, jumlah DFD level 2 sama dengan jumlah modul pada DFD level 1 yang di-breakdown.





4. Membuat DFD Level 3 dan seterusnya

DFD level 3, 4, 5, dan seterusnya merupakan breakdown dari modul pada DFD level di atasnya. Breakdown pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD level 1 atau level 2.

2.4.2 Entity Relationship Diagram (ERD)

Menurut Harsiti (2012:26) *Entity Relationship Diagram (ERD)* adalah diagram yang menghubungkan antar entitas di dalam penyusunan/perancangan basis data. Diagram Entity Relationship ini ditemukan oleh Chen tahun 1976. Tujuan dari Entity Relationship Diagram adalah untuk menunjukkan objek data dan relationship yang ada pada objek tersebut. Berikut komponen dan simbol ERD :

Tabel 2.2 Tabel Komponen dan Simbol ERD

No.	Komponen	Simbol
1.	<i>Entity</i> (entitas)	
2.	<i>Relationship</i> (keterhubungan)	
3.	<i>Atribut</i>	
4.	<i>Link</i> (penghubung antara himpunan relasi dengan himpunan entitas dan penghubung antara himpunan entitas dan atributnya)	

Berikut penjelasan dari komponen elemen-elemen ERD dalam tabel 2.2 :

1. Entitas

Entitas merupakan individu yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Entitas bisa berupa orang, kejadian, atau benda dimana data akan dikumpulkan.

2. Atribut

Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik dari entitas tersebut. Berikut adalah syarat atribut suatu entitas :

- a. Informasi yang diambil tentang sebuah entitas b. Nama atribut harus merupakan kata benda
- c. Kadang nama entitas diletakkan di depan nama atribut untuk ketelitian.

3. Relationship

Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Entitas pertama dalam relationship disebut entitas induk, entitas kedua disebut sebagai entitas anak. Relationship harus mempunyai nama yang berupa kata kerja.

4. Kardinalitas

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas yang lain.

Bentuk relasi antar file dapat dikategorikan menjadi tiga macam. Pengkategorian tersebut dikenal dengan nama cardinality rasio yang menjelaskan tentang batasan pada jumlah entity yang berhubungan melalui relationship. Kardinalitas relasi yang terjadi di antara dua himpunan entitas (misalnya A dan B) dapat berupa :

a. Satu ke Satu (one to one)

Yang berarti setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

Contoh :

A : Jurusan , B : Dosen

Setiap Jurusan hanya bisa dikepalai oleh 1 Dosen.



Gambar 2.2 Contoh kardinalitas *one to one*

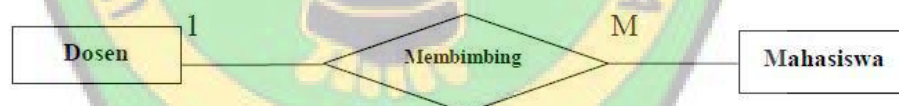
b. Satu ke Banyak (*one to many*)

Yang berarti setiap entitas pada himpunan A dapat berhubungan banyak dengan entitas B, tetapi himpunan entitas B hanya dapat berhubungan dengan paling banyak satu entitas saja pada himpunan entitas A.

Contoh :

A : Dosen , B : Mahasiswa

Setiap dosen bisa membimbing banyak mahasiswa, tapi 1 mahasiswa hanya memiliki 1 dosen pembimbing akademik.



Gambar 2.3 Contoh kardinalitas *one to many*

c. Banyak ke Banyak (*many to many*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

Contoh :

A : Mahasiswa , B : Matakuliah

Setiap mahasiswa bisa mempelajari banyak mata kuliah, sebaliknya setiap Mata Kuliah dapat dipelajari oleh banyak mahasiswa.



Gambar 2.4 Contoh kardinalitas *many to many*

2.4.3 Database

Menurut Ayu (2015), *Database* atau basis data terdiri dari 2 kata, yaitu basis dan data. Basis dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, symbol, teks, gambar, bunyi, atau kombinasinya. Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Basis data dan lemari arsip sesungguhnya memiliki prinsip kerja dan tujuan yang sama. Prinsip utamanya adalah pengaturan data/arsip, dan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data/arsip. Perbedaannya hanya terletak pada media penyimpanan yang digunakan. Jika lemari arsip menggunakan lemari besi atau kayu sebagai media penyimpanan, maka basis data menggunakan media penyimpanan elektronik seperti *disk* (disket atau hard disk). Hal ini merupakan konsekuensi logis, karena lemari arsip langsung ditangani oleh manusia, sementara basis data ditangani melalui perantaraan alat/mesin pintar elektronik (yang dikenal dengan nama komputer). Perbedaan media ini yang selanjutnya melahirkan perbedaan-perbedaan lain yang

menyangkut jumlah dan jenis metoda yang dapat digunakan dalam upaya penyimpanan.

Satu hal yang juga harus diperhatikan, bahwa basis data bukan hanya sekedar penyimpanan data secara elektronis (dengan bantuan komputer). Artinya, tidak semua bentuk penyimpanan data secara elektronis adalah basis data. Penyimpanan dokumen berisi data dalam file teks (dengan program pengolah kata), file *spread sheet*, dan lain-lain tidak dapat disebut sebagai basis data. Hal ini dikarenakan di dalamnya tidak ada pemilihan dan pengelompokkan data sesuai jenis atau fungsi data, sehingga akan menyulitkan pencarian data nantinya.

Pengaturan / pemilihan / pengelompokkan / pengorganisasian data yang disimpan sesuai dengan fungsi dan jenisnya sangat ditonjolkan dalam basis data. Pemilihan/pengelompokkan/pengorganisasian tersebut dapat berbentuk sejumlah file/tabel terpisah atau dalam bentuk pendefinisian kolom-kolom / *field-field* data dalam setiap file atau tabel.

Di dalam sebuah *disk*, basis data dapat diciptakan dan dapat juga ditiadakan. Di dalam sebuah *disk*, dapat ditempatkan beberapa (lebih dari satu) basis data. Sementara dalam sebuah basis data, dapat ditempatkan satu atau lebih file/tabel. Pada file/tabel tersebut data disimpan. Setiap basis data umumnya dibuat untuk mewakili sebuah semesta data yang spesifik. Misalnya, ada basis data kepegawaian, basis data akademik, basis data inventori (pergudangan), dan sebagainya. Contohnya dalam basis data akademik dapat disimpan file mahasiswa, file mata kuliah, file dosen, file jadwal, file nilai, dan seterusnya. Sehingga dapat diketahui operasi-operasi dasar yang berkenaan dengan basis data adalah sebagai berikut:

1. Pembuatan basis data baru (*create database*).
2. Penghapusan basis data (*drop database*).
3. Pembuatan file/tabel baru ke suatu basis data (*create table*).
4. Penghapusan file/tabel dari suatu basis data (*drop table*).
5. Penambahan/pengisian data baru ke sebuah file/tabel di sebuah basis data (*insert*).
6. Pengambilan data dari sebuah file/tabel (*retrieve/search*).
7. Perubahan data dari sebuah file/tabel (*update*).

8. Penghapusan data dari sebuah file/tabel (*delete*).

Operasi yang berkenaan dengan pembuatan objek (basis data dan tabel) merupakan operasi awal yang hanya dilakukan sekali dan berlaku seterusnya. Sedangkan operasi-operasi yang berkaitan dengan isi tabel (data) merupakan operasi rutin yang akan berlangsung berulang-ulang dan oleh karena itu operasi-operasi inilah yang lebih tepat mewakili aktivitas pengelolaan (*management*) dan pengolahan (*processing*) data dalam basis data.

Seperti tertulis di atas bahwa tujuan awal dan utama dalam pengelolaan data dalam sebuah basis data adalah agar dapat memperoleh dan menemukan kembali data dengan mudah dan cepat. Selain pengelolaan data, basis data juga dimanfaatkan untuk tujuan-tujuan lain. Berikut ini adalah pemanfaatan basis data yang dilakukan untuk memenuhi sejumlah tujuan (objektif) :

1. Kecepatan dan kemudahan (*speed*), pemanfaatan basis data dapat memungkinkan untuk dapat menyimpan data atau melakukan perubahan/manipulasi terhadap data atau menampilkan kembali data tersebut dengan lebih cepat dan mudah, dibandingkan dengan menyimpan data secara manual (non elektronik) atau secara elektronik tetapi tidak dalam bentuk penerapan basis data, misalnya dalam bentuk *spread sheet* atau dokumen teks biasa.
2. Efisiensi ruang penyimpanan (*space*), dengan basis data, efisiensi/optimalisasi penggunaan ruang penyimpanan dapat dilakukan, karena dapat dilakukan penekanan jumlah redundansi data, baik dengan menerapkan sejumlah pengkodean atau dengan membuat relasi-relasi (dalam bentuk file) antar kelompok data yang saling berhubungan.
3. Keakuratan (*accuracy*), pemanfaatan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan/batasan (*constraint*) tipe data, domain data, keunikan data, dan sebagainya, yang secara ketat dapat diterapkan dalam sebuah basis data, sangat berguna untuk menekan ketidakakuratan pemasukan/penyimpanan data.
4. Ketersediaan (*availability*), karena kepentingan pemakaian data, sebuah basis data dapat memiliki data yang disebar di banyak lokasi geografis.

Misalnya data nasabah sebuah bank dipisah-pisah dan disimpan di lokasi yang sesuai dengan keberadaan nasabah. Dengan pemanfaatan teknologi jaringan komputer, data yang berada di suatu lokasi atau cabang, dapat juga diakses (menjadi tersedia/*available*) bagi lokasi/cabang lain.

5. Kelengkapan (*completeness*), untuk mengakomodasi kebutuhan kelengkapan data yang semakin berkembang, maka tidak hanya dengan menambah *record - record* data, tetapi juga melakukan perubahan struktur dalam basis data, baik dalam bentuk penambahan objek baru (tabel) atau dengan penambahan *field- field* baru pada suatu tabel.
6. Keamanan (*security*), pengaksesan terhadap basis data dapat ditentukan mengenai siapa saja yang boleh menggunakan basis data beserta objek-objek di dalamnya dan penentuan terhadap jenis-jenis operasi apa saja yang boleh dilakukan.
7. Kebersamaan pemakaian (*sharebility*), penggunaan basis data tidak terbatas oleh satu pengguna saja, atau di satu lokasi tertentu atau oleh satu aplikasi/sistem saja. Tetapi basis data dapat dimanfaatkan untuk mendukung lingkungan multiuser.

2.4.4 MySQL

Alexander (2011:49-50) menyatakan *MySQL* atau dibaca “*My Sekuel*” adalah suatu *RDBMS (Relational Database Management System)* yaitu aplikasi sistem yang menjalankan fungsi pengolahan data. *MySQL* pertama dikembangkan oleh *MySQL AB* yang kemudian diakusisi *Sun Microsystem* dan terakhir dikelola oleh *Oracle Cooperation*.

MySQL adalah *database* yang sangat populer digunakan bersama *PHP*. *PHP* dengan *MySQL* adalah kombinasi yang baik untuk menampilkan kekuatan sesungguhnya dari *Server-Side scripting*. Dengan *MySQL*, pengguna dapat membuat *table*, dimana data dapat disimpan lebih efisien dibandingkan menyimpan data dalam *array*. Untuk menggunakan *MySQL* ataupun *database* lain secara efektif, diperlukan pemahaman *Structured Query Language (SQL)*.

2.4.5 XAMPP

Muhammad Lutfi Indrawan (2013:3-4) *XAMPP* adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*Localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* merupakan singkatan dari *X* (sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

2.4.5.1 Asal Kata XAMPP

Muhammad Lutfi Indrawan (2013:3-4) *XAMPP* adalah singkatan yang masing-masing hurufnya adalah :

1. *X* : Program ini dapat dijalankan dibanyak sistem operasi, seperti *Windows*, *Linux*, *Mac OS* dan *Solaris*.
2. *A* : *Apache*, merupakan aplikasi *web server*. Tugas utama *Apache* adalah menghasilkan halaman *web* yang benar kepada user berdasarkan kode *PHP* yang dituliskan oleh pembuat halaman *web*. jika diperlukan juga berdasarkan kode *PHP* yang dituliskan, maka dapat saja suatu *database* diakses terlebih dahulu (Misalnya, dalam *MySQL*) untuk mendukung halaman *web* yang dihasilkan
3. *M* : *MySQL*, merupakan aplikasi *database server*. Perkembangannya disebut *SQL* yang merupakan kepanjangan dari *Structured Query Language*. *SQL* merupakan bahasa terstruktur yang digunakan untuk mengolah *database*. *MySQL* dapat digunakan untuk membuat dan mengelola *database* beserta isinya. Dapat memanfaatkan *MySQL* untuk menambahkan, mengubah, dan menghapus data yang berada dalam *database*.
4. *P* : *PHP*, bahasa pemrograman *web*. Bahasa pemrograman *PHP* merupakan bahasa pemrograman untuk membuat *web* yang bersifat *server-side scripting*. *PHP* memungkinkan kita untuk membuat halaman *web* yang

bersifat dinamis. Sistem manajemen basis data yang sering digunakan bersama *PHP* adalah *MySQL*. Namun *PHP* juga mendukung sistem manajemen *database Oracle*, *Microsoft Access*, *Interbase*, *d-base*, *PostgreSQL* dan sebagainya.

5. *P* : *Perl* adalah bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin *Unix*. *Perl* dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya *Perl 1*. Pada *versi-versi* selanjutnya, *Perl* tersedia pula untuk berbagai sistem operasi varian *Unix* (*SunOS*, *Linux*, *BSD*, *HP-UX*), juga tersedia untuk sistem operasi seperti *DOS*, *Windows*, *PowerPC*, *BeOS*, *VMS*, *EBCDIC* dan *PocketPC*. Dukungan terhadap pemrograman berbasis obyek (*Object Oriented Programming* atau *OOP*) ditambahkan pada *Perl 5*, yang pertama kali dirilis pada tanggal 31 Juli 1993. Proyek pengembangan *Perl 6* dimulai pada tahun 2000.

2.4.6 Web Browser dan HTML

Wahyu T. Sapro (2017) *Web browser* adalah program untuk menampilkan halaman yang berbentuk kode HTML. Semua halaman *web* ditulis dengan bahasa HTML (*Hypertext Mark Up Language*). Walaupun beberapa file mempunyai ekstensi yang berbeda (contoh *.html*, *.php*, *.php3*), output file-file tersebut tetap HTML. HTML adalah medium yang selalu dikirimkan ke *web browser* baik halaman itu berupa halaman statis, sebuah script (seperti PHP), ataupun yang dibuat oleh program *CGI* (*Common Gateway Interface*).

2.4.7 PHP

Menurut Syakir Rahman (2014), pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs Personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (Form Interpreted), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data form dari *Web*. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilsan kode sumber ini

menjadi open source, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP dirubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi *Web* kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

2.4.7.1 Pengertian PHP

Anwar Budiarto (2015) menyatakan *PHP* adalah singkatan dari *PHP Hypertext Preprocessor*, merupakan bahasa berbentuk skrip yang ditempatkan dalam *server* dan diproses di *server*. Kemudian hasilnya dikirimkan ke klien, tempat pemakai menggunakan browser. Secara khusus PHP dirancang untuk membentuk aplikasi *Web* dinamis, artinya dapat menampilkan data terbaru sesuai permintaan klien pada *browser*.

Konsep kerja PHP berawal dari model kerja HTML (*Hypertext Markup Language*) yang diawali dengan permintaan suatu halaman *Web* oleh *browser*. Berdasarkan URL (*Uniform Resource Locator*) atau dikenal dengan sebutan alamat internet, *browser* mendapatkan alamat dari *Web server*, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan

oleh *Web server*. Selanjutnya, *Web server* akan mencari file yang diminta dan memberikan isinya ke *Web browser*. *Browser* yang mendapatkan isinya segera melakukan proses penerjemahan kode HTML, dan menampilkannya ke layar pemakai.

Prinsip kerja PHP tidak jauh berbeda dengan prinsip kerja HTML. Perbedaannya pada saat berkas PHP yang diminta didapatkan oleh *Web server*, isinya segera dikirimkan ke mesin PHP dan mesin inilah yang memproses dan memberikan hasilnya (berupa kode HTML) dan menyampaikannya kepada klien.

Salah satu kelebihan PHP adalah mampu berkomunikasi dengan berbagai basis data populer. Sehingga, implementasi penampilan data yang bersifat dinamis, yang diambil dari basis data dapat dilakukan dengan mudah. Itulah sebabnya PHP sering dikatakan sangat cocok untuk membangun *Web* dinamis. PHP dapat dikomunikasikan dengan banyak jenis basis data, diantaranya adalah Microsoft Access, MS SQL, MySQL, Oracle, PostgreSQL, dan Sybase.

2.4.8 Content Management System (CMS)

Yuhefizar. Dkk (2011:7) menyatakan salah satu media dalam interaksi manusia dan komputer adalah interface. *Pengguna interface* sebagai salah satu bagian dari sebuah perangkat lunak mempunyai peranan penting dalam membantu kita untuk mengerti apa yang terjadi di antara pengguna dan sistem juga menerjemahkan tujuan dan apa yang diinginkan oleh *Pengguna* serta apa yang harus dikerjakan oleh sistem.

Interface harus dirancang agar memenuhi prinsip *usability* (kegunaan) namun juga tidak mengurangi nilai ergonomiknya. Kini ragam *interface* pengguna sudah semakin banyak, salah satunya adalah *iconic interface* yang menggunakan gambar sebagai simbol. Seiring perubahan zaman dimana aplikasi berbasis *Web* lebih mendominasi dan menyamai aplikasi desktop dari segala hal termasuk *interface*. Memungkinkan *interface* ini untuk diterapkan pada aplikasi berbasis *Web*.

Aplikasi yang banyak menggunakan *Iconic Interface* ini adalah aplikasi *content Management System (CMS)*. Aplikasi seperti ini banyak dimiliki oleh

berbagai situs, seperti situs jejaring sosial, portal berita, *e-learning* dan masih banyak lagi.

Modul *Content Management System* digunakan untuk melakukan modifikasi *Website*, sehingga *content Web site* menjadi lebih dinamis sesuai yang anda inginkan. *Content Management System* adalah sebuah software yang berfungsi untuk mengatur *content* dari suatu situs. Menambah *content*, mengubah *content*, dan menghapus *content* menjadi lebih mudah. Tampilannya sederhana sehingga mudah dipahami.

Fitur-fitur yang dimiliki oleh *Web Content Manajemen System (CMS)* adalah:

1. Dapat dengan mudah menambah, mengubah, menghapus konten *Web*.
2. Dapat mengupload gambar dengan mudah.
3. Memiliki WYSIWYG Ubahor (mirip MS Word). Yaitu mengubah halaman *Web* semudah mengubah teks di Microsoft Word. Tampilan/format dari MS Word dapat langsung di *copy paste* dan tampilan *Web* pun akan menjadi sama.
4. Hak admin dapat dikelompokkan menjadi beberapa grup. Misal: group admin berita, group admin forum, group admin mengunduh, dll.
5. Melakukan koneksi tabel dengan *Web server* secara langsung (*on the fly*).

2.4.9 Cascading Style Sheet (CSS)

Alexander (2011:112) menyatakan *cascading style sheet* dikembangkan untuk menata gaya pengaturan halaman web. Pada awalnya CSS dikembangkan pada SGML pada tahun 1970, dan terus dikembangkan hingga saat ini CSS telah mendukung banyak bahasa Markup, seperti HTML, XHTML, XML, SVG (Scalable Vector Graphics) dan Mozilla XUL (XML User Inteface Language). Mengacu arti bahasa, *Cascading Style Sheet* memiliki arti Gaya Menata Halaman Bertingkat, yang berarti setiap satu elemen yang telah diformat dan memiliki anak dan telah diformat, maka anak dari semua elemen tersebut secara otomatis mengikuti format elemen induknya.

Pada bulan Desember tahun 1996 W3C mengenalkan spesifikasi CSS level 1 atau juga dikenal CSS yang mendukung format huruf, warna pada teks dan lain-

lainnya. Kemudian pada bulan Mei tahun 1988 W3C mengeluarkan CSS2 yang didalamnya terdapat fungsi pengaturan letak elemen, dan saat ini W3C telah memperbaiki serta meningkatkan kemampuan CSS2 menjadi CSS3. Cascading Style Sheet terdiri dari selector, property dan nilai. Seperti halnya HTML, PHP dan bahasa pemrograman lainnya, CSS juga memiliki aturan penulisan. Berikut ini aturan penulisan CSS.

2.4.10 JQuery

Menurut Gede Erik Aktama (2014), JQuery adalah librari JavaScript yang memungkinkan kita untuk membuat program *Web* pada suatu halaman *Web*, tanpa harus secara eksplisit kita menambahkan event atau pun properti pada halaman *Web* tersebut.

Dengan JQuery, suatu halaman *Web* yang menjadi aplikasi *Web*, jika dilihat sourcenya, akan terlihat seperti dokumen HTML biasa, tidak ada kode JavaScript yang terlihat langsung. Teknik pemrograman *Web* seperti ini disebut sebagai *unobstrusive JavaScript programming*.

JQuery merupakan salah satu librari yang membuat program *Web* di sisi klien, tidak terlihat sebagai program JavaScript biasa, yang harus secara eksplisit disisipkan pada dokumen *Web*. Pada teknik pemrograman sisi klien dengan menggunakan JavaScript biasa, setiap elemen yang akan memiliki event, akan secara eksplisit terlihat ada event yang dilekatkan pada elemen tersebut.

JQuery dikembangkan pertama kali oleh John Resig, yang dibuat lebih sederhana dari librari Prototype yang menjadi inspirasi dari librari JQuery ini. Secara pemrograman, JQuery memiliki kemiripan seperti Prototype.

Lisensi JQuery adalah *open source* yang bisa diperoleh secara cuma-cuma, dan dapat digunakan untuk kepentingan komersial, tanpa ada tuntutan untuk membayar kepada pembuat JQuery. Lisensi opensource dari JQuery adalah *Massachusetts Institute of Technology (MIT)* dan *GNU - Lesser General Public License (LGPL)*.

2.4.11 *Asynchronous Javascript and XML (AJAX)*

Menurut Asep Sufyan (2008), sejak tahun 2005 Jesse James Garret dari Adaptive Path telah memperkenalkan sebuah teknologi baru yang merupakan gabungan antara ((X)HTML, CSS, JavaScript, *Document Object Model (DOM)* dan *XML Http Request*) yang disebut AJAX. Dengan teknologi AJAX yang disebut “*AJAX Engine*”, proses yang dilakukan oleh *AJAX Engine* adalah menerima respon dari server, melakukan *parsing* (penguraian) data dan melakukan sedikit perubahan untuk menampilkan informasi. Oleh karena proses mengirimkan sedikit informasi, jika dibandingkan dengan model aplikasi *Web* tradisional maka *user interface* diperbaharui lebih cepat sehingga user dapat menyelesaikan pekerjaannya dengan cepat. AJAX banyak diterapkan dalam teknologi *Web* saat ini karena AJAX memiliki banyak keistimewaan yang cocok diterapkan pada kebutuhan aplikasi *Web* saat ini. Diantaranya keistimewaan AJAX adalah:

1. Aplikasi dapat melakukan *request* ke server tanpa harus melakukan pergantian halaman.
2. Lebih cepat dan hemat bandwidth, karena aplikasi tidak perlu melakukan *reloading* terhadap seluruh halaman. Hanya konten yang diperlukan saja yang akan di *load*.
3. Proses terjadi secara *asynchronous* dibelakang layar.

Dari keistimewaan-keistimewaan AJAX inilah dapat dihasilkan aplikasi *Web* yang semakin interaktif dan dinamis. Sehingga selagi AJAX melakukan *request* dan menunggu respon, *user* tetap dapat berinteraksi dengan antar muka aplikasi. Aplikasi *Web* konvensional dengan aplikasi *Web* menggunakan AJAX memiliki perbedaan arsitektur.

Selain itu, yang membedakan antara *Web* AJAX dengan *Web* konvensional biasa adalah kemampuannya bekerja di belakang layar secara *asynchronous*. Berikut adalah gambaran prinsip kerja *Synchronous* dan *Asynchronous* AJAX dibandingkan dengan *Web* konvensional biasa.

User berinteraksi dengan link-link yang ada untuk meminta proses ke server, proses yang diminta akan dikirimkan secara *background* ke server, selama server belum mengembalikan data hasil proses, *user* dapat tetap bekerja dengan

halaman *Web* yang ada sebelumnya. Setelah server selesai melakukan proses, hasil proses tersebut akan dikirimkan kembali kepada *Web* browser, saat data yang dikirimkan server telah diterima pada *Web* browser *user* maka data tersebut ditampilkan pada halaman *Web* yang telah ada sebelumnya.

Tahap 4 : Construction (Konstruksi)

Construction merupakan proses membuat kode. Coding atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Programmer akan menterjemahkan kebutuhan user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan aplikasi, artinya penggunaan computer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem dan uji coba terhadap pengguna melalui kuesioner. Tujuan testing dan uji coba terhadap pengguna adalah untuk menemukan kesalahan-kesalahan terhadap sistem dan mengetahui sistem yang dibuat layak digunakan atau tidak agar kemudian bisa diperbaiki. Untuk menterjemahkan kode-kode program menggunakan bahasa pemrograman PHP & MySQL dan pengujian dilakukan menggunakan Blackbox Testing serta uji coba terhadap pengguna dengan menyebarkan kuesioner yang kemudian diproses untuk dihitung menggunakan skala likert.

2.4.12 Testing (Black Box)

Pengujian *Black Box* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian *black box* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dieksekusi pada perangkat lunak dan kemudian keluar dari perangkat lunak dicek apakah telah sesuai yang diharapkan (Rizky, 2011). Pengujian *Black Box* berusaha menemukan kesalahan dalam kategori:

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.

4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

2.4.13 *Skala Likert*

Pengertian atau definisi Skala Likert adalah skala yang digunakan untuk mengukur persepsi, sikap atau pendapat seseorang atau kelompok mengenai sebuah peristiwa atau fenomena sosial, berdasarkan definisi operasional yang telah ditetapkan oleh peneliti. Skala ini merupakan suatu skala psikometrik yang biasa diaplikasikan dalam angket dan paling sering digunakan untuk riset yang berupa survei, termasuk dalam penelitian survei deskriptif.

Penggagas dan pencipta skala likert adalah Rensis Likert asal Amerika Serikat yang menerbitkan suatu laporan yang menjelaskan penggunaannya. Skala Likert digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Dengan Skala Likert, variabel yang akan diukur dijabarkan menjadi indikator variabel. Dalam pengukuran bidang pendidikan, skala Likert juga sering digunakan, selain juga skala Guttman, semantik Diferensial, Rating scale, dan skala Thurstone. Bentuk jawaban skala Likert antara lain: sangat setuju, setuju, ragu-ragu, tidak setuju, dan tidak setuju. Selain itu, jawaban setiap item instrumen yang menggunakan Skala Likert bisa juga mempunyai gradasi dari sangat positif sampai sangat negatif, yang dapat berupa kata-kata antara lain: Sangat Penting (SP), Penting (P), Ragu-ragu (R), Tidak Penting (TP), Sangat Tidak Penting (STP).

Tahap 5 : *Deployment* (Penyerahan)

Tahapan ini merupakan tahap akhir /final dalam pembuatan sebuah software atau sistem. Setelah melakukan analisis, desain dan pengkodean maka system yang sudah jadi akan digunakan oleh user. Kemudian software yang telah dibuat harus evaluasi jika ada kekurangan dan dilakukan pemeliharaan secara berkala.

Akan tetapi, pada pembuatan aplikasi ini penulis hanya melakukan sampai pada tahapan testing (pengujian).

BAB III

ANALISI DAN DESAIN

3.1. *Communication* (Komunikasi)

Pada tahap ini, melakukan observasi dan komunikasi langsung dengan Wakil Kepala Sekolah Bagian Kesiswaan SMAN 1 Kapuas Hilir. Komunikasi yang dilakukan berupa konsultasi dan wawancara. Adapun hasil dari observasi dilapangan dan konsultasi, yaitu berupa data yang dibutuhkan.

Data yang dibutuhkan dalam pembuatan Aplikasi Penjurusan di SMAN 1 Kapuas Hilir Berbasis *Web* yaitu data siswa, data nilai raport dan data pedoman penerimaan peserta didik baru. Data-data tersebut didapatkan dari Wakil Kepala Sekolah Bagian Kesiswaan SMAN 1 Kapuas Hilir.

3.1.1. *Teknologi*

Penggunaan teknologi sangat erat kaitannya dengan proses pembuatan Aplikasi Penjurusan di SMAN 1 Kapuas Hilir Berbasis *Web* ini. Secara umum tahapan analisa teknologi ini yaitu analisa teknologi perangkat keras yang digunakan dan perangkat lunak/aplikasi yang dipergunakan untuk membuat desain dan koding yang dipergunakan dalam merancang dan membangun website ini.

3.1.1.1. *Hardware* / Perangkat Keras

Spesifikasi perangkat keras yang digunakan adalah notebook Acer D270 dengan spesifikasi:

- a. Processor : Intel(R) Atom (TM) N2600
- b. RAM : 2 GBDDR3

3.1.1.2. *Software* / Perangkat Lunak

Perangkat lunak merupakan alat pendukung sistem yang terdiri dari sistem operasi dan aplikasi-aplikasi lainnya. Adapun perangkat lunak yang digunakan adalah sebagai berikut:

- a. Sistem Operasi : Microsoft Windows 7 Ultimate
- b. *MySQL* sebagai *database*

- c. *XAMPP* sebagai web server
- d. *Notepad++* sebagai editor *coding*
- e. *Google Chrome* sebagai *web browser* yang digunakan
- f. *EdrawMax* untuk permodelan pada alat metodologi *Waterfall Modelling*.
- g. *Adobe Photoshop* sebagai editor gambar

3.2. *Planning*(Perencanaan)

Proses *planning* merupakan lanjutan dari proses *communication* (komunikasi). Tahapan ini akan menghasilkan data yang berhubungan dengan keinginan *user* dalam pembuatan *software*, termasuk rencana yang akan dilakukan. Berikut merupakan proses bisnis sistem lama penjurusan siswa dan rekomentasi sistem baru pembuatan Aplikasi Penjurusan di SMAN 1 Kapuas Hilir berbasis *Web*.

3.2.1. Analisis Sistem Lama

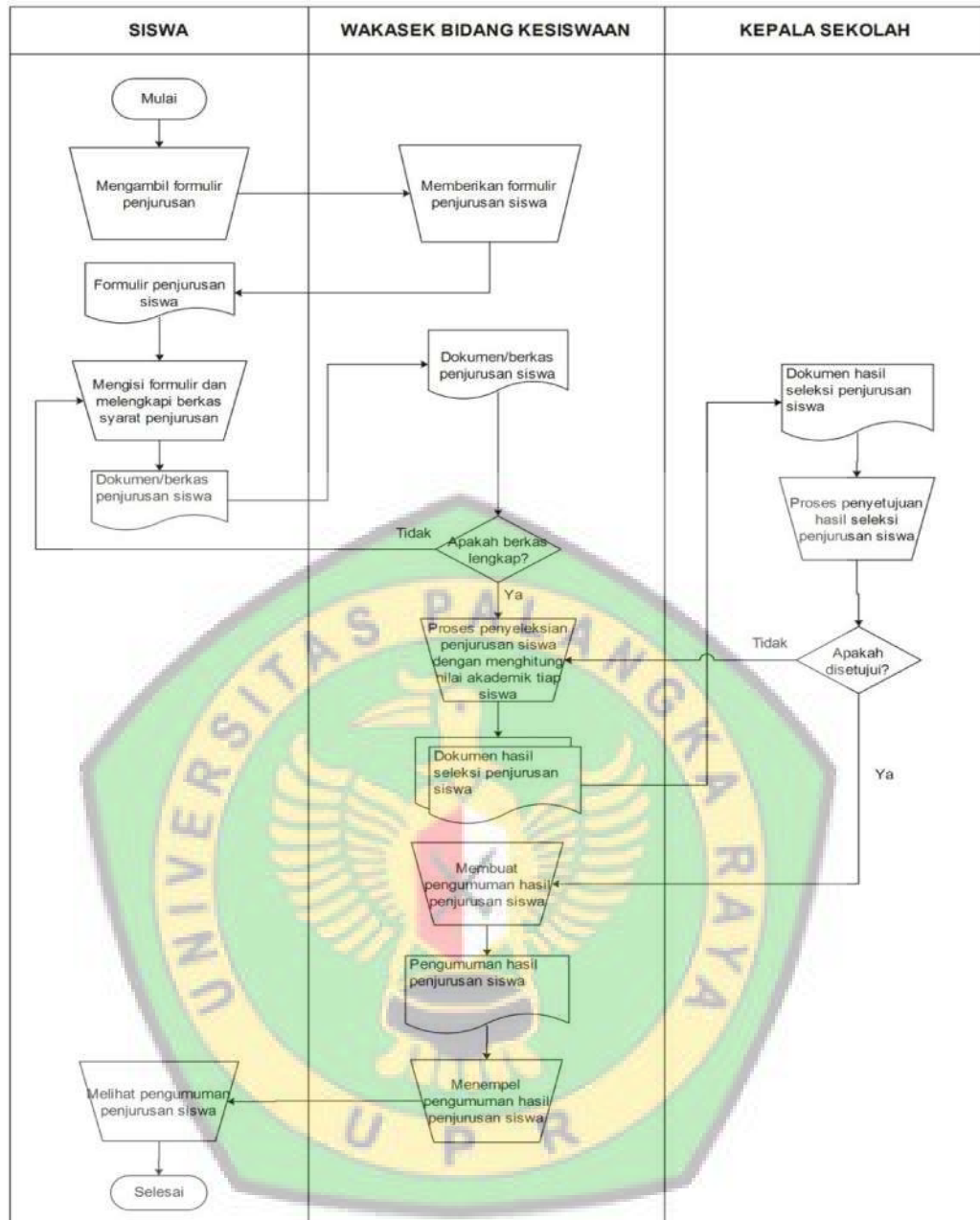
Diperlukan ketepatan dalam penentuan pemilihan jurusan karena akan berdampak pada masa depan siswa. Akan tetapi dalam penentuan jurusan siswa ini masih kurang efektif karena dalam penyeleksian penjurusan siswa hanya dengan menghitung nilai akademik sedangkan masih ada beberapa kriteria lain yang juga harus dipertimbangkan dalam penentuan jurusan. Pada proses penyeleksian penjurusan ini memiliki kelemahan karena membutuhkan waktu cukup lama terutama pada saat menghitung nilai akademik masing-masing siswa yang cukup banyak bisa terjadi banyak kekeliruan karena belum tersedianya sistem khusus untuk mendukung perhitungan tersebut. Kemudian pada proses pengambilan formulir penjurusan, siswa diharuskan datang ke sekolah untuk mendatangi wakasek bidang kesiswaan untuk mengambil formulir penjurusan, hal ini juga membutuhkan waktu cukup lama.

Dari hasil analisa dan pengamatan, maka dapat ditentukan deskripsi sistem lama adalah sebagai.

3.2.1.1. Deskripsi Sistem Lama

1. Siswa mengambil formulir penjurusan yang dapat diambil pada Wakasek Bidang Kesiswaan.
2. Wakasek Bidang Kesiswaan memberikan formulir penjurusan kepada siswa kelas X SMAN 1 Kapuas Hilir.
3. Siswa mengisi formulir dan melengkapi berkas syarat penjurusan.
4. Siswa mengembalikan formulir dan berkas penjurusan ke Wakasek Bidang Kesiswaan.
5. Jika berkas dan data lengkap maka diproses dan diseleksi, tetapi jika belum lengkap maka akan dikembalikan ke siswa untuk dilengkapi.
6. Proses penyeleksian jurusan hanya berdasarkan satu kriteria yaitu dengan cara perhitungan nilai akademik siswa yang dilakukan oleh Wakasek Bidang Kesiswaan.
7. Wakasek Bidang Kesiswaan menyerahkan laporan hasil seleksi penjurusan siswa untuk disetujui.
8. Kepala Sekolah menyetujui laporan hasil seleksi penjurusan siswa, jika tidak disetujui maka akan dikembalikan ke Wakasek Bidang Kesiswaan untuk diseleksi penjurusan kembali.
9. Wakasek Bidang Kesiswaan mendapatkan persetujuan Kepala Sekolah tentang hasil penjurusan siswa.
10. Wakasek Bidang Kesiswaan membuat pengumuman hasil penjurusan siswa
11. Wakasek Bidang Kesiswaan menempel pengumuman di papan pengumuman hasil penjurusan siswa.
12. Siswa melihat pengumuman.

Flowchart Sistem Lama yaitu seperti Gambar 3.1..



Gambar 3.1. Flowchart Sistem Lama

3.2.1.2. Kesimpulan Sistem Lama

Dari sistem lama yang masih berjalan, terdapat kelemahan yaitu sebagai.

1. Penentuan penjurusan siswa pada sistem yang berjalan saat ini bisa dikatakan masih belum kurang efektif, karena hanya melihat satu kriteria saja yaitu dengan cara perhitungan nilai akademik siswa yang dilakukan oleh Wakasek Bidang Kesiswaan, sebenarnya masih ada kriteria lain yang harus dipertimbangkan dalam mengambil keputusan penentuan jurusan siswa.

Kemudian Wakasek Bidang Kesiswaan membutuhkan waktu cukup lama dalam proses penyeleksian jurusan terutama pada saat menghitung nilai akademik tiap siswa, sedangkan banyak jumlah siswa yang harus ditentukan jurusannya.

2. Pada proses pengambilan formulir penjurusan, siswa diharuskan ke sekolah untuk mendatangi wakasek bidang kesiswaan untuk mengambil formulir penjurusan.

3.2.2. Analisi Sistem Baru

Dari analisis sistem lama yang sudah dibahas telah menjelaskan tentang sistem penyeleksian penjurusan siswa masih dikatakan kurang efektif karena hanya melihat satu kriteria saja yaitu dengan cara perhitungan nilai akademik siswa, kemudian dalam perhitungan nilai akademik siswa juga membutuhkan waktu cukup lama karena harus menghitung satu per satu sedangkan banyak jumlah siswa yang harus ditentukan jurusannya. Kemudian pada proses pengambilan formulir penjurusan, siswa diharuskan datang ke sekolah untuk mendatangi wakasek bidang kesiswaan untuk mengambil formulir penjurusan, hal ini juga membutuhkan waktu cukup lama. Hal ini menjadi alasan untuk dibuatnya aplikasi penjurusan siswa di SMAN 1 Kapuas Hilir menggunakan metode SMART berbasis *web* yang diharapkan dapat membantu SMAN 1 Kapuas Hilir untuk menentukan penjurusan tiap siswa.

3.2.2.1. Rekomendasi Sistem Baru

Rekomendasi Sistem Baru yang direkomendasikan dalam menyikapi kelemahan sistem lama ini adalah:

1. Merancang dan membangun Aplikasi Penjurusan Siswa berbasis *Web* untuk menentukan penjurusan tiap siswa di SMAN 1 Kapuas Hilir sesuai kriteria yang ditentukan dan diakses secara online agar siswa tidak lagi datang ke sekolah untuk mengambil formulir. Pada sistem penunjang keputusan ini akan dibuat menggunakan Metode *Simple Multy Attribute Rating Technique (SMART)* yang akan menyelesaikan masalah dengan kriteria dan bobot yang

ditentukan. Dalam sistem akan dilakukan proses perhitungan sehingga sistem akan otomatis menentukan penjurusan tiap siswa.

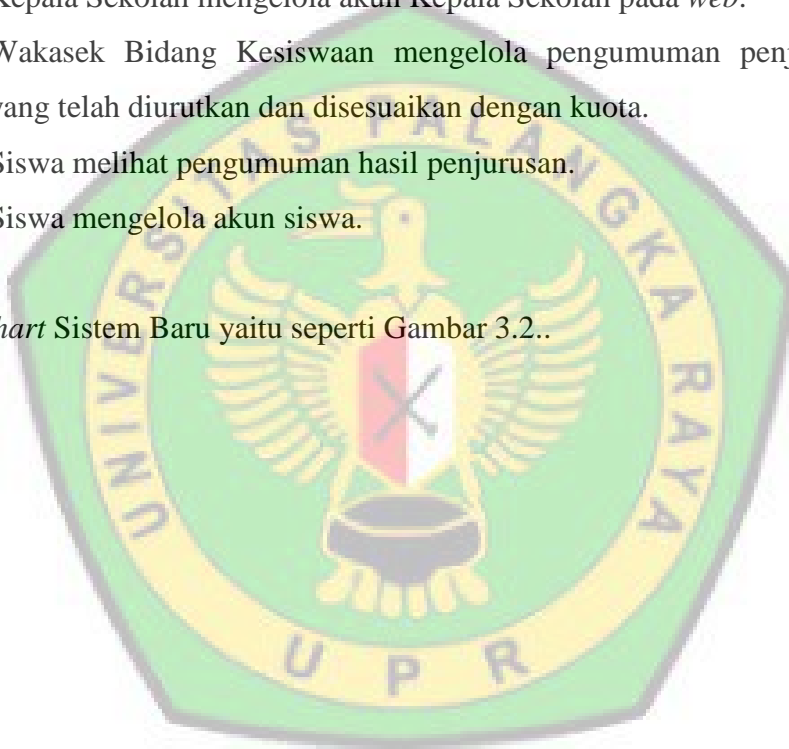
3.2.2.2. Deskripsi Sistem Baru

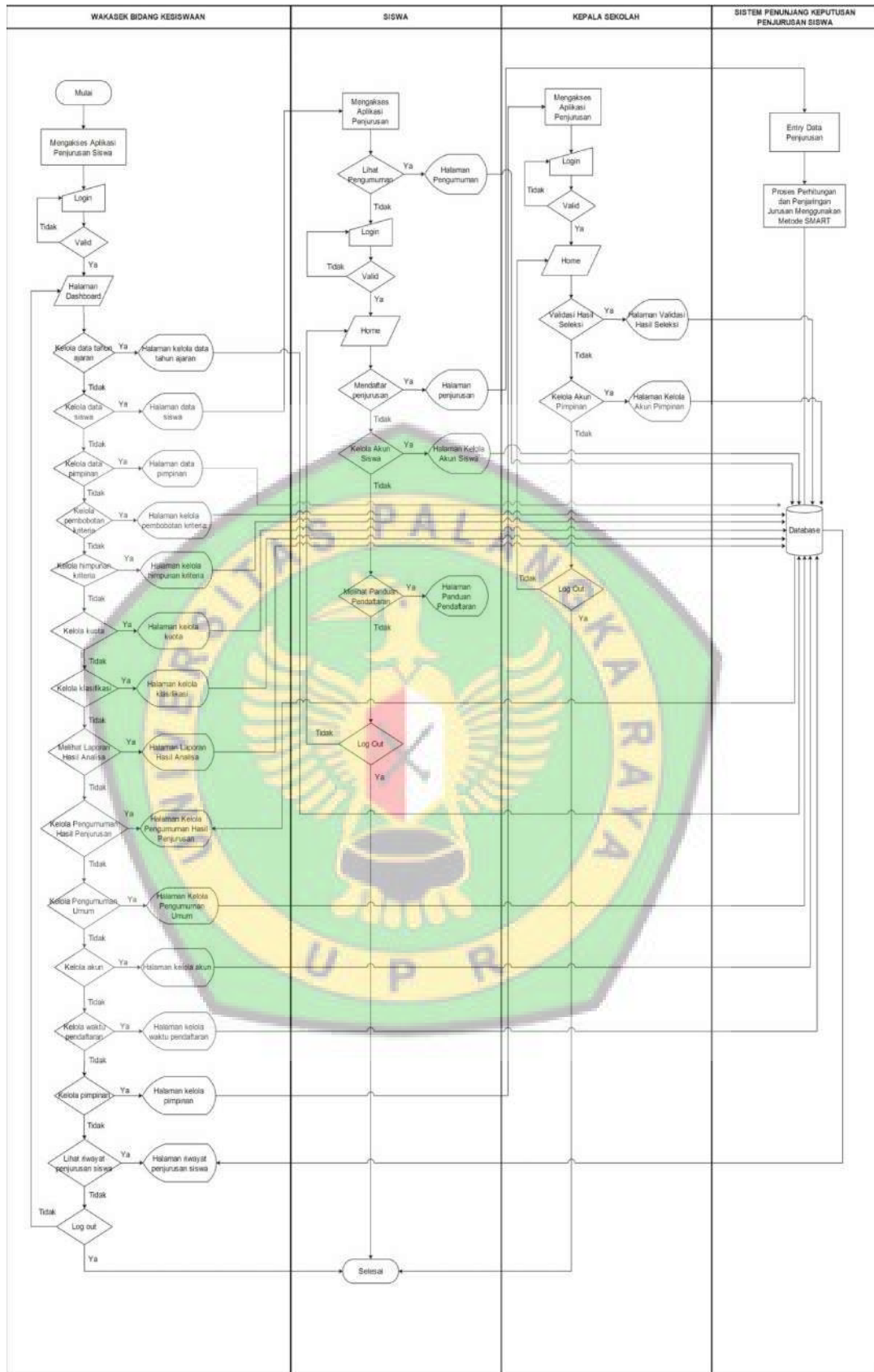
Pada aplikasi sistem penunjang keputusan penjurusan di SMAN 1 Kapuas Hilir berbasis *web* in terdapat 3 pengguna yaitu Siswa, Wakasek Bidang Kesiswaan dan Kepala Sekolah. Berikut adalah bisnis proses sistem baru.

1. Wakasek Bidang Kesiswaan mengakses aplikasi penjurusan siswa berbasis *web*.
2. Wakasek Bidang Kesiswaan *login* pada *website*. Jika *login* gagal, maka akan kembali ke halaman *login*. Jika *login* berhasil maka akan masuk ke Halaman *Dashboard*.
3. Wakasek Bidang Kesiswaan mengelola data tahun ajaran.
4. Wakasek Bidang Kesiswaan mengelola data siswa ke sistem sehingga siswa mendapatkan *username* dan *password* untuk mengakses dan mendaftar penjurusan melalui *website*.
5. Wakasek Bidang Kesiswaan mengelola data pimpinan yang berwenang mengelola penjurusan siswa sehingga pimpinan mendapatkan *username* dan *password* jika ingin mengakses *website*.
6. Wakasek Bidang Kesiswaan mengelola pembobotan kriteria.
7. Wakasek Bidang Kesiswaan mengelola himpunan kriteria.
8. Wakasek Bidang Kesiswaan mengelola kuota penjurusan.
9. Wakasek Bidang Kesiswaan mengelola waktu pendaftaran.
10. Wakasek Bidang Kesiswaan mengelola pengumuman umum.
11. Wakasek Bidang Kesiswaan melihat riwayat penjurusan siswa per-tahun ajaran.
12. Wakasek Bidang Kesiswaan mengelola akun Wakasek Bidang Kesiswaan.
13. Siswa mengakses *website* penjurusan menggunakan *username* dan *password*.
14. Siswa melihat panduan pendaftaran penjurusan pada *website*.
15. Siswa mendaftar penjurusan di *website*.

16. Wakasek Bidang Kesiswaan melihat dan mengelola data klasifikasi tiap siswa.
17. Sistem menganalisa hasil penjurusan tiap siswa.
18. Wakasek Bidang Kesiswaan melihat laporan hasil analisa penjurusan yang dianalisa menggunakan metode SMART.
19. Kepala Sekolah mengakses aplikasi penjurusan berbasis *web*.
20. Kepala Sekolah login pada *website*. Jika login gagal, maka akan kembali ke halaman login. Jika login berhasil maka akan masuk ke Home.
21. Kepala Sekolah validasi hasil penjurusan.
22. Kepala Sekolah mengelola akun Kepala Sekolah pada *web*.
23. Wakasek Bidang Kesiswaan mengelola pengumuman penjurusan siswa yang telah diurutkan dan disesuaikan dengan kuota.
24. Siswa melihat pengumuman hasil penjurusan.
25. Siswa mengelola akun siswa.

Flowchart Sistem Baru yaitu seperti Gambar 3.2..





Gambar 3.2. Flowchart Sistem Baru

3.2.2.3. Kesimpulan Sistem Baru

Setelah melakukan analisis sistem maka akan dibangun rancangan sistem baru untuk mengatasi kekurangan di sistem lama. Berikut adalah kesimpulan dari sistem baru.

1. Pengguna

Pengguna yang ada di *website* ini yaitu sebagai.

- a. Wakasek Bidang Kesiswaan
- b. Siswa
- c. Kepala Sekolah

2. Aktivitas Pengguna

Aktivitas yang pengguna/aktor lakukan yaitu sebagai.

- a. Wakasek Bidang Kesiswaan
 - 1) Login (Data login dan info login)
 - 2) Kelola tahun ajaran (Data tahun ajaran dan info tahun ajaran)
 - 3) Kelola data siswa (Data siswa dan info siswa)
 - 4) Kelola data pimpinan (Data pengguna dan info pengguna)
 - 5) Kelola pembobotan kriteria (Data pembobotan kriteria dan info pembobotan kriteria)
 - 6) Kelola himpunan kriteria (Data himpunan kriteria dan info himpunan kriteria)
 - 7) Kelola kuota (Data kuota dan Info kuota)
 - 8) Kelola klasifikasi (Data klasifikasi dan info klasifikasi)
 - 9) Lihat laporan hasil analisa (Info hasil analisa)
 - 10) Kelola pengumuman hasil penjurusan (Data hasil seleksi dan Info hasil seleksi)
 - 11) Kelola Pengumuman Umum (Data pengumuman dan Info pengumuman)
 - 12) Kelola waktu pendaftaran (Data pendaftaran dan Info pendaftaran)
 - 13) Lihat riwayat penjurusan (Info siswa, info klasifikasi, info tahun ajaran)
 - 14) Kelola akun wakasek bidang kesiswaan (Data pengguna dan info pengguna)

- b. Siswa
- 1) Melihat pengumuman umum (Info pengumuman)
 - 2) Melihat pengumuman hasil penjurusan (Info hasil seleksi)
 - 3) Login (Data login dan info login)
 - 4) Mendaftar Penjurusan (Data klasifikasi, data berkas, info berkas dan info klasifikasi)
 - 5) Kelola akun siswa (Data siswa dan info siswa)
 - 6) Melihat panduan pendaftaran (Info panduan pendaftaran)
- c. Kepala Sekolah
- 1) Login (Data login dan info login)
 - 2) Validasi hasil seleksi (Data hasil seleksi dan info hasil seleksi)
 - 3) Kelola akun kepala sekolah (Data pengguna dan info pengguna)

Berikut merupakan rencana kerja yang akan dilakukan untuk pembuatan aplikasi yang dalam hal ini adalah Aplikasi Penjurusan Siswa di SMAN 1 Kapuas Hilir Berbasis *Web* pada Tabel 3.1.

Tabel 3.1 Rencana Kegiatan

No	Kegiatan	Estimasi Waktu Pelaksanaan
1	Communication(Komunikasi)	2 Bulan
	Observasi dan konsultasi untuk mengumpulkan kebutuhan	
2	Planning(Perencanaan)	
	Menganalisis Kebutuhan dan penyusunan rencana kerja	
3	Modeling(Permodelan)	
	Perhitungan dengan menggunakan Metode <i>SMART</i>	
	Permodelan desain program dalam bentuk diagram	
	Perancangan desain antarmuka program	
4	Construction(Konstruksi)	
	Desain antarmuka program	
	Penulisan kode program(<i>coding</i>)	

3.3 Modeling(Permodelan)

Proses modeling ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan software yang dapat diperkirakan sebelum dibuat coding. Proses ini terbagi menjadi 2, yaitu analisis dan desain. Pada analisis menggunakan Data Flow Diagram (DFD) serta Entity Relationship Diagram namun sebelum memasuki tahapan ini, user terlebih dahulu menganalisis metode yang digunakan untuk penjurusan yang dilakukan di SMAN 1 Kapus Hilir secara manual pada microsoft excel menggunakan metode Simple Multi Attribute Rating Technique (SMART).

Pada desain akan mendesain tabel, desain navigasi menggunakan Sitemap dan desain antarmuka/representasi interface. Tahapan ini akan menghasilkan dokumen yang disebut software requirement.

3.3.1 Analisis

3.3.1.1 Perhitungan Metode SMART (Simple Multi Attribute Rating Technique)

Langkah-langkah yang dilakukan dalam proses penjurusan menggunakan metode *Simple Multy Attribute Rating* (SMART) adalah sebagai.

Langkah 1 : Menentukan jumlah kriteria.

Tabel 3.2 Tabel Kriteria, Sub Kriteria dan Nilai Sub Kriteria

No.	Kriteria	Subkriteria	Nilai
1	Nilai Rata-rata Ajaran	81 - 100	4
		71 - 80	3
		61 - 70	2
		45 - 60	1
2	Nilai Rata-rata IPA	81 - 100	4
		71 - 80	3
		61 - 70	2
		45 - 60	1

3	Nilai Rata-rata IPS	81 - 100	1
		71 - 80	2
		61 - 70	3
		45 - 60	4
4	Prestasi dibidang IPA	Internasional / Nasional	4
		Propinsi	3
		Kabupaten / Kecamatan	2
		Tidak Ada	1
5	Minat Siswa	IPA	4
		IPS	1

Langkah 2 : Sistem secara default memberikan skala 0 – 100 berdasarkan prioritas yang telah diinputkan kemudian dilakukan normalisasi.

Rumus Normalisasi adalah : $\frac{W_j}{\sum W_j}$

Keterangan :

W_j : bobot suatu kriteria

Tabel 3.3 Tabel Normalisasi

No.	Kriteria	Bobot	Normalisasi
1	Nilai Rata-rata Ajaran	30	$\frac{30}{30+25+25+10+10} = 0,3$
2	Nilai Rata-rata IPA	25	$\frac{25}{30+25+25+10+10} = 0,25$
3	Nilai Rata-rata IPS	25	$\frac{25}{30+25+25+10+10} = 0,25$
4	Prestasi dibidang IPA	10	$\frac{10}{30+25+25+10+10} = 0,1$
5	Minat Siswa	10	$\frac{10}{30+25+25+10+10} = 0,1$
Total		100	1

Langkah 3 : Memberikan nilai kriteria untuk setiap alternatif.

Tabel 3.4 Tabel Nilai Kriteria di Setiap Alternatif

Alternatif / Kriteria	K1	K2	K3	K4	K5
<i>Budi</i>	4	4	4	1	4
<i>Ani</i>	4	1	4	1	4
<i>Dodo</i>	3	4	3	1	1

Langkah 4 : Menghitung nilai utility untuk setiap kriteria masing-masing.

$$u_i(a_i) = 100 \frac{(C_{out\ i} - C_{min})}{(C_{max} - C_{min})} \%$$

Keterangan :

$u_i(a_i)$: nilai utility kriteria ke – 1 untuk kriteria ke – i

C_{max} : nilai kriteria maxsimal

C_{min} : nilai kriteria minimal

$C_{out\ i}$: nilai kriteria ke – i

Tabel 3.5 Tabel menentukan nilai utility

No.	Alternatif	Kriteria	Utility
1	Budi	K1	$u_i(a_i) = 100 \frac{(4-1)}{(4-1)} = 100,00$
		K2	$u_i(a_i) = 100 \frac{(4-1)}{(4-1)} = 100,00$
		K3	$u_i(a_i) = 100 \frac{(4-1)}{(4-1)} = 100,00$
		K4	$u_i(a_i) = 100 \frac{(1-1)}{(4-1)} = 0,00$
		K5	$u_i(a_i) = 100 \frac{(4-1)}{(4-1)} = 100,00$
2	Ani	K1	$u_i(a_i) = 100 \frac{(4-1)}{(4-1)} = 100,00$
		K2	$u_i(a_i) = 100 \frac{(1-1)}{(4-1)} = 0,00$
		K3	$u_i(a_i) = 100 \frac{(4-1)}{(4-1)} = 100,00$
		K4	$u_i(a_i) = 100 \frac{(1-1)}{(4-1)} = 0,00$
		K5	$u_i(a_i) = 100 \frac{(4-1)}{(4-1)} = 100,00$

No.	Alternatif	Kriteria	Utility
3	Dodo	K1	$u_i(a_i) = 100 \frac{(1-1)^{0,3}}{(4-1)} = 66,67$
		K2	$u_i(a_i) = 100 \frac{(1-1)^{0,25}}{(4-1)} = 100,00$
		K3	$u_i(a_i) = 100 \frac{(1-1)^{0,25}}{(4-1)} = 66,67$
		K4	$u_i(a_i) = 100 \frac{(1-1)^{0,1}}{(4-1)} = 0,00$
		K5	$u_i(a_i) = 100 \frac{(1-1)^{0,25}}{(4-1)} = 0,00$

Langkah 5 : Menghitung nilai akhir masing-masing.

$$u(a_i) = \sum_{j=1}^m w_j u_i(a_i)$$

Keterangan :

$u(a_i)$: nilai total alternatif

w_j : hasil dari normalisasi bobot kriteria

$u_i(a_i)$: hasil penentuan nilai utility

$$\text{Budi} = (0,3 \times 100,00) + (0,25 \times 100,00) + (0,25 \times 100,00) + (0,1 \times 0,00) + (0,1 \times 100,00) = 90,00$$

$$\text{Ani} = (0,3 \times 100,00) + (0,25 \times 100,00) + (0,25 \times 100,00) + (0,1 \times 0,00) + (0,1 \times 100,00) = 65,00$$

$$\text{Dodo} = (0,3 \times 100,00) + (0,25 \times 100,00) + (0,25 \times 100,00) + (0,1 \times 0,00) + (0,1 \times 100,00) = 61,67$$

Tabel. 3.6 Tabel Penentuan Nilai Total Masing-Masing Alternatif

No.	Alternatif	Kriteria	Utility	Bobot Normalisasi	Skor / Nilai Akhir
1	Budi	K1	100,00	0,3	90,00
		K2	100,00	0,25	
		K3	100,00	0,25	
		K4	0,00	0,1	
		K5	100,00	0,1	

2	Ani	K1	100,00	0,3	65,00
		K2	0,00	0,25	
		K3	100,00	0,25	
		K4	0,00	0,1	
		K5	100,00	0,1	
3	Dodo	K1	66,67	0,3	61,67
		K2	100,00	0,25	
		K3	66,67	0,25	
		K4	0,00	0,1	
		K5	0,00	0,1	

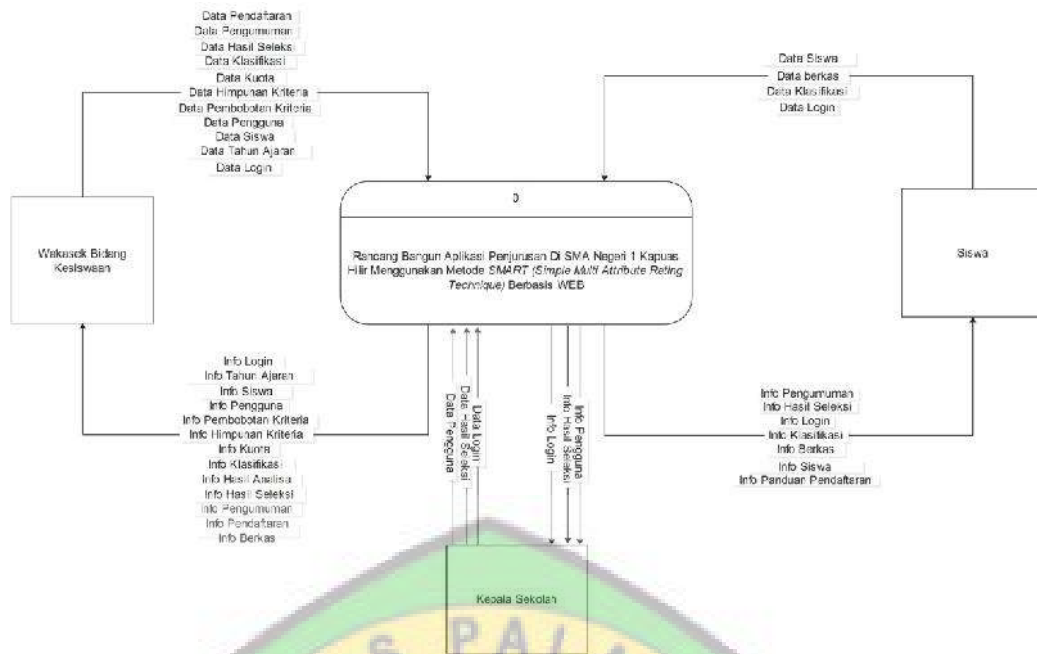
Jadi jika nilai masing-masing siswa sudah diurutkan dari yang terbesar ke terkecil, yang kemudian nilai tersebut akan difilter berdasarkan kuota yang telah ditentukan.

3.3.1.2 *Data Flow Diagram (DFD)*

DFD ini akan menjelaskan bagaimana sistem yang akan dibuat nantinya dan menjelaskan aliran data dari satu proses ke proses yang lainnya melalui beberapa tahapan seperti.

1. **Diagram Konteks**

Diagram Konteks digunakan untuk menggambarkan sistem perangkat lunak untuk membuat Aplikasi Penjurusan di SMA Negeri 1 Kapuas Hilir Menggunakan Metode *SMART (Simple Multi Attribute Rating Technique)* Berbasis *WEB* secara garis besar atau keseluruhan. Diagram Konteks ini dirancang memperhatikan masukan yang dibutuhkan oleh sistem dan keluaran yang dihasilkan oleh sistem. Diagram Konteks dapat dilihat pada gambar 3.3.



Gambar 3.3 Diagram Konteks Aplikasi Penjurusan

2. Tabel Entitas

Definisi entitas digunakan untuk menjelaskan data apa saja yang masuk dan keluar pada sistem. Berikut ini adalah Tabel entitas perancangan perangkat lunak untuk membuat Aplikasi Penjurusan di SMA Negeri 1 Kapuas Hilir Menggunakan Metode *SMART (Simple Multi Attribute Rating Technique)* Berbasis *WEB* pada Tabel 4.2.

Tabel 3.7 Entitas

No	Entitas luar	Keterangan
1	Wakasek Bidang Kesiswaan	<p>Pengguna yang mengakses dan mengelola Sistem Penunjang Keputusan Penjurusan Siswa Berbasis <i>Web</i> di SMA Negeri 1 Kapuas Hilir secara keseluruhan.</p> <p><u>INPUT :</u></p> <ol style="list-style-type: none"> 1. Data Login 2. Data Tahun Ajaran 3. Data Siswa

No	Entitas luar	Keterangan
		<p>4. Data Pengguna</p> <p>5. Data Pembobotan Kriteria</p> <p>6. Data Himpunan Kriteria</p> <p>7. Data Kuota</p> <p>8. Data Klasifikasi</p> <p>9. Data Hasil Seleksi</p> <p>10. Data Pengumuman</p> <p>11. Data Pendaftaran</p> <p><u>OUTPUT:</u></p> <p>1. Info Login</p> <p>2. Info Tahun Ajaran</p> <p>3. Info Siswa</p> <p>4. Info Pengguna</p> <p>5. Info Pembobotan Kriteria</p> <p>6. Info Himpunan Kriteria</p> <p>7. Info Kuota</p> <p>8. Info Klasifikasi</p> <p>9. Info Hasil Analisa</p> <p>10. Info Hasil Seleksi</p> <p>11. Info Pengumuman</p> <p>12. Info Pendaftaran</p> <p>13. Info Berkas</p>
2	Siswa	<p>Pengguna yang dapat mendaftar penjurusan siswa dengan mengakses <i>web</i> Aplikasi Penjurusan, dimana siswa dapat mendaftar, mengelola akun dan yang dapat mendaftar dan melihat pengumuman.</p> <p><u>INPUT :</u></p> <p>1. Data Login</p> <p>2. Data Klasifikasi</p> <p>3. Data Berkas</p> <p>4. Data Siswa</p>

No	Entitas luar	Keterangan
		<p><u>OUTPUT:</u></p> <ol style="list-style-type: none"> 1. Info Pengumuman 2. Info Hasil Seleksi 3. Info Login 4. Info Klasifikasi 5. Info Berkas 6. Info Siswa 7. Info Panduan Pendaftaran
3	<p>Kepala Sekolah</p>	<p>Pengguna yang bertugas memvalidasi atau menyetujui hasil seleksi Penjurusan Siswa.</p> <p><u>INPUT :</u></p> <ol style="list-style-type: none"> 1. Data Login 2. Data Hasil Seleksi 3. Data Pengguna <p><u>OUTPUT:</u></p> <ol style="list-style-type: none"> 1. Info Login 2. Info Hasil Seleksi 3. Info Pengguna

3. DFD Level-1

Penggambaran sistem DFD level-1 merupakan penjabaran dari diagram konteks, hanya pada level ini sudah menjurus kepada suatu proses dan merupakan gabungan secara keseluruhan yang melibatkan semua kesatuan luar secara lengkap. DFD level-1 perancangan perangkat lunak untuk membuat Aplikasi Penjurusan di SMA Negeri 1 Kapuas Hilir Menggunakan Metode *SMART (Simple Multi Attribute Rating Technique)* Berbasis *WEB* dapat dilihat pada gambar 3.4.

4. Definisi *Storage*

Definisi *Storage* digunakan untuk menjelaskan tabel apa saja yang digunakan dalam sistem. Berikut ini adalah Tabel definisi *storage* dalam membuat Aplikasi Penjurusan Siswa SMAN 1 Kapuas Hilir berbasis *web* pada Tabel 3.8.

Tabel 3.8 Definisi *Storage*

No	Nama Penyimpanan	Keterangan
1.	Tabel pengguna	Tabel yang digunakan untuk tempat menyimpan data akun wakasek bidang kesiswaan dan data akun pimpinan.
2.	Tabel siswa	Tabel yang digunakan untuk tempat menyimpan data siswa.
3.	Tabel pengumuman	Tabel yang digunakan untuk tempat menyimpan data pengumuman umum tentang penjurusan.
4.	Tabel himpunankriteria	Tabel yang digunakan untuk tempat menyimpan data himpunan, keterangan serta nilai/rating yang ada pada suatu kriteria.
5.	Tabel klasifikasi	Tabel yang digunakan untuk tempat menyimpan data klasifikasi tiap siswa.
6.	Tabel kriteria	Tabel yang digunakan untuk tempat menyimpan data kriteria dan bobot untuk penilaian penjurusan siswa.
7.	Tabel kuota	Tabel yang digunakan untuk tempat menyimpan kuota penjurusan siswa.
8.	Tabel berkas	Tabel yang digunakan untuk tempat menyimpan berkas siswa.
9.	Tabel waktupendaftaran	Tabel yang digunakan untuk tempat menyimpan pengaturan waktu pendaftaran penjurusan siswa.
10.	Tabel ranking	Tabel yang digunakan untuk menyimpan data hasil seleksi penjurusan siswa.

No	Nama Penyimpanan	Keterangan
11	Tabel tahun_ajaran	Tabel yang digunakan untuk menampung data tahun ajaran penjurusan siswa.

5. Definisi Proses

Definisi Proses digunakan untuk menjelaskan proses apa saja yang digunakan dalam sistem, serta menjelaskan *input* maupun *output* yang digunakan dalam proses tersebut. Berikut ini adalah Tabel Definisi Proses dalam pembuatan Aplikasi Penjurusan Siswa SMAN 1 Kapuas Hilir berbasis *web* pada Tabel 3.9.

Tabel 3.9 Definisi Proses

No	Nama Proses	Input	Output	Keterangan
1	Login	Data login: 1. <i>Username</i> 2. <i>Password</i>	Info login: 1. Login sukses 2. Login gagal	Proses validasi akun wakasek bidang kesiswaan, siswa dan pimpinan, digunakan untuk menjaga keamanan sistem
2	Kelola Data	Data: 1. Klik sub-menu Kelola Data Pimpinan 2. Klik sub-menu Kelola Data Siswa 3. Klik sub-menu Kelola Data Tahun Ajaran	Info Data: 1. Info halaman Kelola Data Pengguna 2. Info halaman Kelola Data Siswa 3. Info halaman Kelola Data Tahun Ajaran	Proses untuk mengelola Data yang terdiri dari Kelola Data Siswa, Kelola Data Pengguna dan Kelola Data Tahun Ajaran
3	Kelola SPK Penjurusan	Data SPK Penjurusan: 1. Klik sub-menu Kelola	Info SPK Penjurusan: 1. Info halaman Kelola	Proses untuk mengelola Data SPK Penjurusan yang terdiri dari Kelola

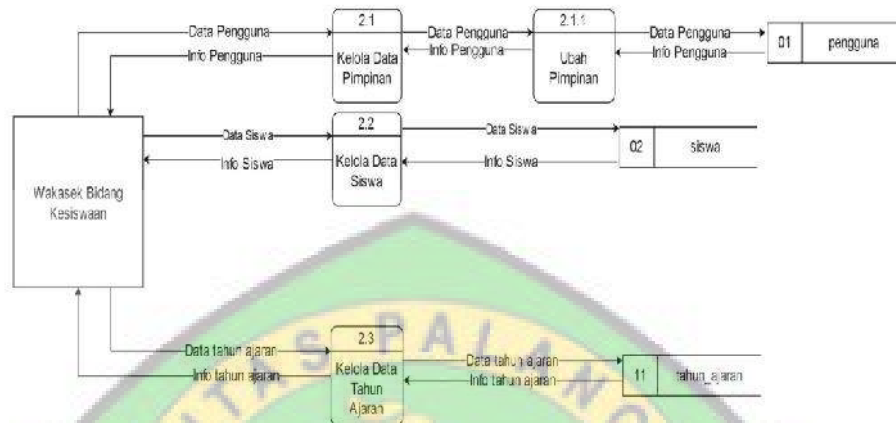
No	Nama Proses	Input	Output	Keterangan
		Pembobotan Kriteria 2. Klik sub-menu Kelola Himpunan Kriteria 3. Klik sub-menu Kelola Klasifikasi 4. Klik sub-menu Kelola Kuota	Pembobotan Kriteria 2. Info halaman Kelola Himpunan Kriteria 3. Info halaman Kelola Klasifikasi 4. Info halaman Kelola Kuota	Pembobotan Kriteria, Kelola Himpunan Kriteria, Kelola Klasifikasi dan Kelola Kuota
4	Melihat Hasil Analisa	Klik menu Laporan Hasil Analisa	Info Laporan Hasil Analisa	Melihat laporan analisa SMART
5	Kelola Pengumuman Penjurusan	Klik menu Pengumuman Penjurusan	Info Halaman Kelola Pengumuman Penjurusan	Proses mengelola pengumuman penjurusan
6	Kelola Pengumuman Umum	Data Pengumuman: 1. Tambah pengumuman 2. Ubah pengumuman 3. Hapus pengumuman	Info Pengumuman: 1. Pengumuman berhasil ditambahkan 2. Pengumuman berhasil diubah 3. Pengumuman berhasil dihapus	Proses untuk mengelola data pengumuman umum
7	Kelola Akun Wakasek Bidang	Data Akun Wakasek Bidang Kesiswaan: 1. Ubah akun	Info Akun Wakasek Bidang Kesiswaan:	Proses untuk mengelola atau mengubah akun

No	Nama Proses	Input	Output	Keterangan
	Kesiswaan	Wakasek Bidang Kesiswaan	1. Akun berhasil diubah	Wakasek Bidang Kesiswaan
8	Kelola Waktu Pendaftaran	Data Waktu Pendaftaran: 1. Ubah waktu pendaftaran	Info Waktu Pendaftaran: 1. Waktu Pendaftaran berhasil diubah	Proses untuk mengelola atau mengubah waktu pendaftaran
9	Lihat Riwayat Penjurusan	Klik menu Riwayat Penjurusan	Info Halaman Riwayat Penjurusan	Proses untuk melihat riwayat penjurusan
10	Lihat Pengumuman	Klik menu Pengumuman	Info pengumuman umum dan pengumuman hasil seleksi	Proses untuk melihat pengumuman umum dan pengumuman hasil seleksi
11	Mendaftar penjurusan	Data Mendaftar Penjurusan: 1. Klik Daftar	Info Mendaftar Penjurusan: 1. Info halaman Pendaftaran	Proses untuk mendaftar penjurusan
11	Kelola Akun Siswa	Data Akun Siswa: 1. Ubah akun	Info Akun Siswa: 1. Akun berhasil diubah	Proses untuk mengelola atau mengubah akunsiswa
12	Validasi Hasil Seleksi	Klik menu Validasi Hasil Seleksi	Info Halaman Validasi Hasil Seleksi	Proses untuk memvalidasi/ menyetujui hasil seleksi
13	Kelola Akun Kepala Sekolah	Data Akun Kepala Sekolah: 1. Ubah akun	Info Akun Kepala Sekolah: 1. Akun berhasil diubah	Proses untuk mengelola atau mengubah akun Kepala Sekolah

6. DFD Level 2

6.1 DFD Level 2 Proses 2.0 (Kelola Data)

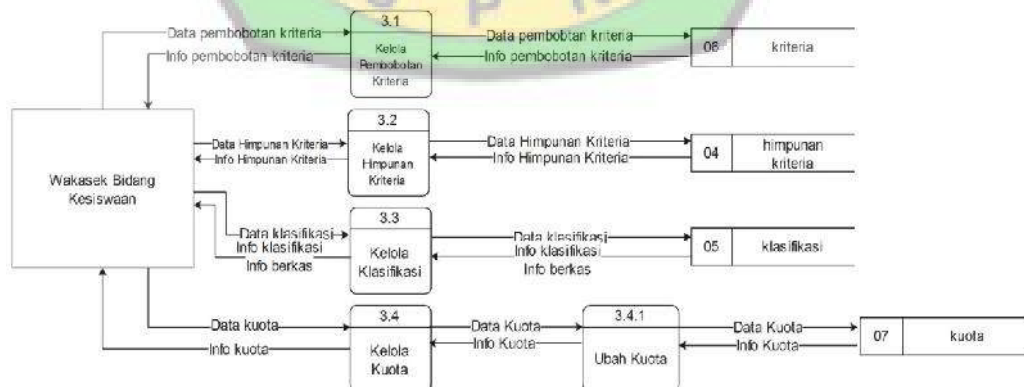
DFD Level-2 Wakasek Bidang Kesiswaan Proses 2 (Kelola Data) didalamnya terdapat kelola data pimpinan, kelola data siswa dan kelola data tahun ajaran. DFD Level-2 Proses 2.0 yaitu seperti pada Gambar 3.5.



Gambar 3.5. DFD Level 2 Proses 2.0

6.2 DFD Level 2 Proses 3.0 (Kelola Data SPK Penjurusan)

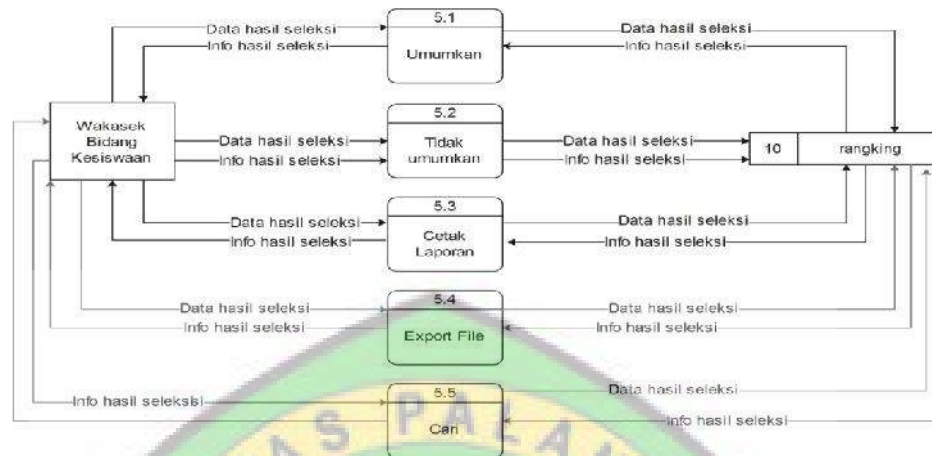
DFD Level-2 Wakasek Bidang Kesiswaan Proses 3.0 (Kelola Data SPK Penjurusan) didalamnya terdapat kelola pembobotan kriteria, kelola himpunan kriteria, kelola klasifikasi dan kelola kuota. DFD Level-2 Proses 3.0 yaitu seperti pada Gambar 3.6.



Gambar 3.6. DFD Level 2 Proses 3.0

6.3 DFD Level 2 Proses 5.0 (Kelola Pengumuman Penjurusan)

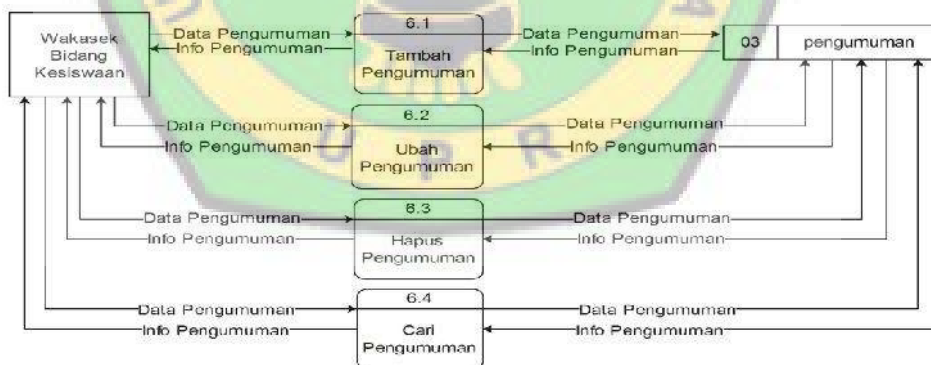
DFD Level-2 Wakasek Bidang Kesiswaan Proses 5.0 (Kelola Pengumuman Penjurusan) didalamnya terdapat *button* cetak, export file, umumkan dan tidak umumkan. DFD Level-2 Proses 5.0 yaitu seperti pada Gambar 3.7.



Gambar 3.7. DFD Level 2 Proses 5.0

6.4 DFD Level 2 Proses 6.0 (Kelola Pengumuman Umum)

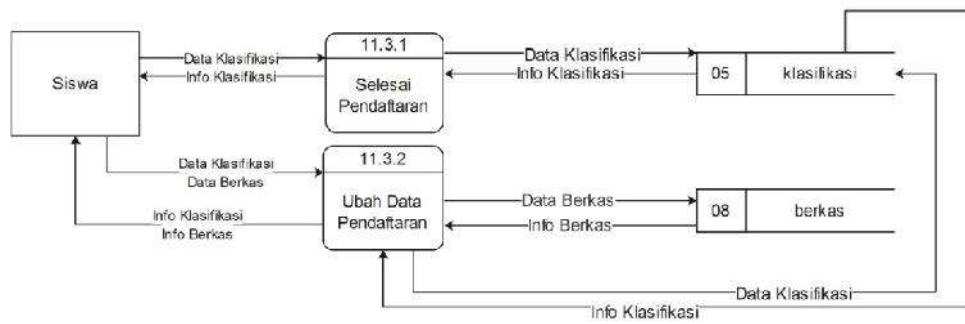
DFD Level-2 Wakasek Bidang Kesiswaan Proses 6.0 (Kelola Pengumuman Umum) terdapat proses tambah, ubah, hapus dan cari. DFD Level-2 Proses 6.0 yaitu seperti pada Gambar 3.8.



Gambar 3.8. DFD Level 2 Proses 6.0

6.5 DFD Level 2 Proses 11.1 (Verifikasi)

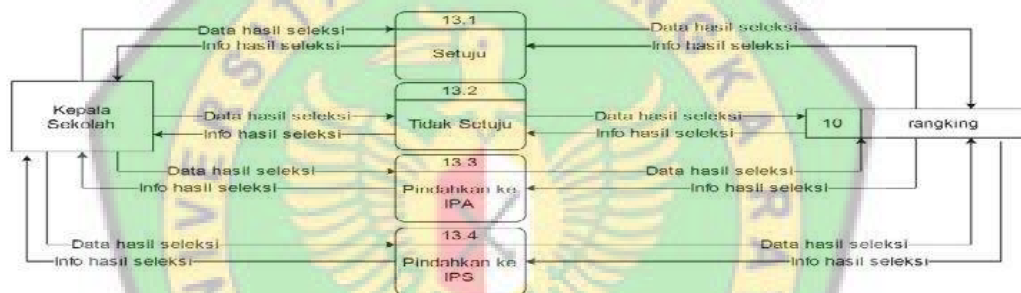
DFD Level-2 Siswa Proses 11.1 (Verifikas) didalamnya terdapat proses selesai dan ubah. DFD Level-2 Proses 11.1 yaitu seperti pada Gambar 3.9.



Gambar 3.9. DFD Level 2 Proses 11.1

6.6 DFD Level 2 Proses 13.0 (Validasi Hasil Seleksi)

DFD Level-2 Kepala Sekolah Proses 13.0 (Validasi Hasil Seleksi) didalamnya terdapat proses setuju, tidak setuju, pindahkan ke IPA dan pindahkan ke IPS. DFD Level-2 Proses 13.0 yaitu seperti pada Gambar 3.10.

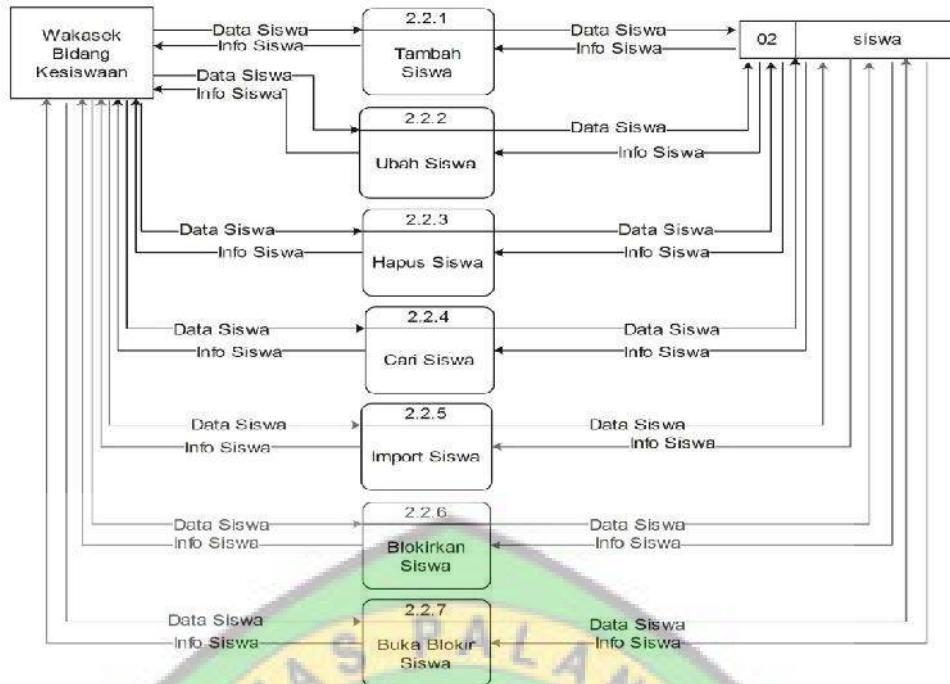


Gambar 3.10. DFD Level 2 Proses 13.0

7. DFD Level 3

7.1 DFD Level 3 Proses 2.2 (Kelola Data Siswa)

DFD Level-3 Wakasek Bidang Kesiswaan Proses 2.2 (Kelola Data Siswa) didalamnya terdapat proses tambah, ubah, hapus, cari, import, detail, blokirkan dan buka blokir siswa. DFD Level-3 Proses 2.2 yaitu seperti pada Gambar 3.11.



Gambar 3.11. DFD Level 3 Proses 2.2

7.2 DFD Level 3 Proses 2.3 (Kelola Data Tahun Ajaran)

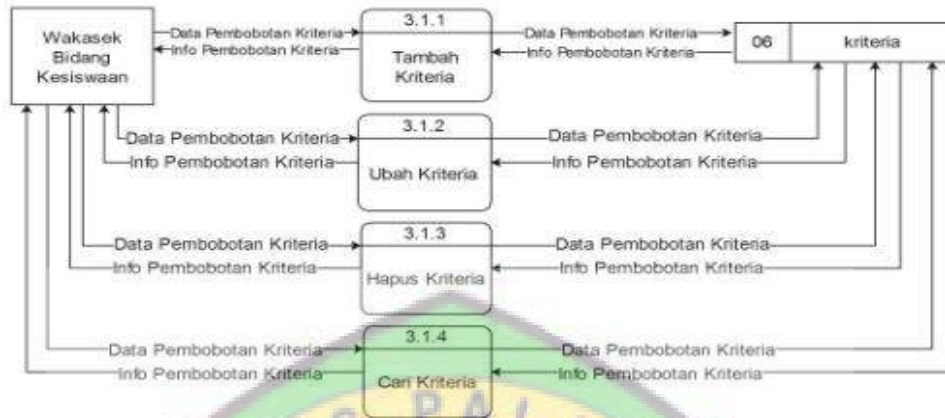
DFD Level-3 Wakasek Bidang Kesiswaan Proses 2.3 (Kelola Data Tahun Ajaran) didalamnya terdapat proses tambah, ubah, hapus dan cari tahun ajaran. DFD Level-3 Proses 2.3 yaitu seperti pada Gambar 3.12.



Gambar 3.12. DFD Level 3 Proses 2.3

7.3 DFD Level 3 Proses 3.1 (Kelola Pembobotan Kriteria)

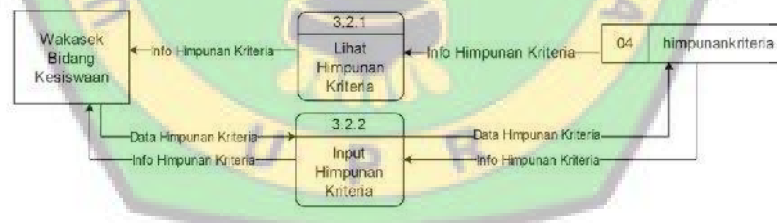
DFD Level-3 Wakasek Bidang Kesiswaan Proses 3.1 (Kelola Pembobotan Kriteria) didalamnya terdapat proses tambah, ubah, hapus dan cari pembobotan kriteria. DFD Level-3 Proses 3.1 yaitu seperti pada Gambar 3.13.



Gambar 3.13. DFD Level 3 Proses 3.1

7.4 DFD Level 3 Proses 3.2 (Kelola Himpunan Kriteria)

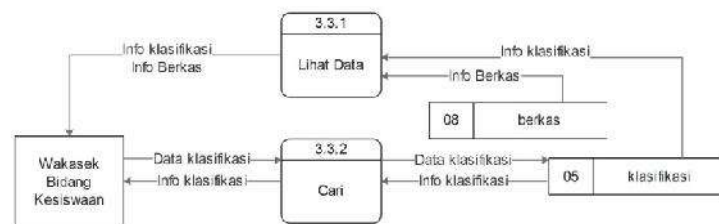
DFD Level-3 Wakasek Bidang Kesiswaan Proses 3.2 (Kelola Himpunan Kriteria) didalamnya terdapat proses lihat himpunan kriteria dan proses input himpunan kriteria. DFD Level-3 Proses 3.2 yaitu seperti pada Gambar 3.14.



Gambar 3.14. DFD Level 3 Proses 3.2

7.5 DFD Level 3 Proses 3.3 (Kelola Klasifikasi)

DFD Level-3 Wakasek Bidang Kesiswaan Proses 3.3 (Kelola Klasifikasi) didalamnya terdapat proses lihat dan cari data. DFD Level-3 Proses 3.3 yaitu seperti pada Gambar 3.15.



Gambar 3.15. DFD Level 3 Proses 3.3

8. DFD Level 4

8.1 DFD Level 4 Proses 3.2.2 (Input Himpunan Kriteria)

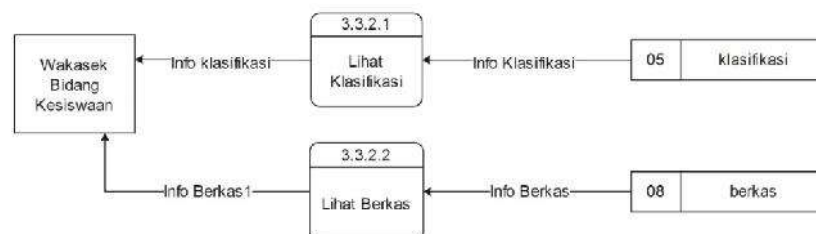
DFD Level-4 Wakasek Bidang Kesiswaan Proses 3.2.2 (Input Himpunan Kriteria) didalamnya terdapat proses tambah, ubah, hapus dan cari himpunan kriteria. DFD Level-4 Proses 3.2.2 yaitu seperti pada Gambar 3.16.



Gambar 3.16. DFD Level 4 Proses 3.2.2

8.2 DFD Level 4 Proses 3.3.2 (Lihat Data)

DFD Level-4 Wakasek Bidang Kesiswaan Proses 3.3.2 (Lihat Data) didalamnya terdapat proses lihat klasifikasi dan lihat berkas. DFD Level-4 Proses 3.3.2 yaitu seperti pada Gambar 3.17.

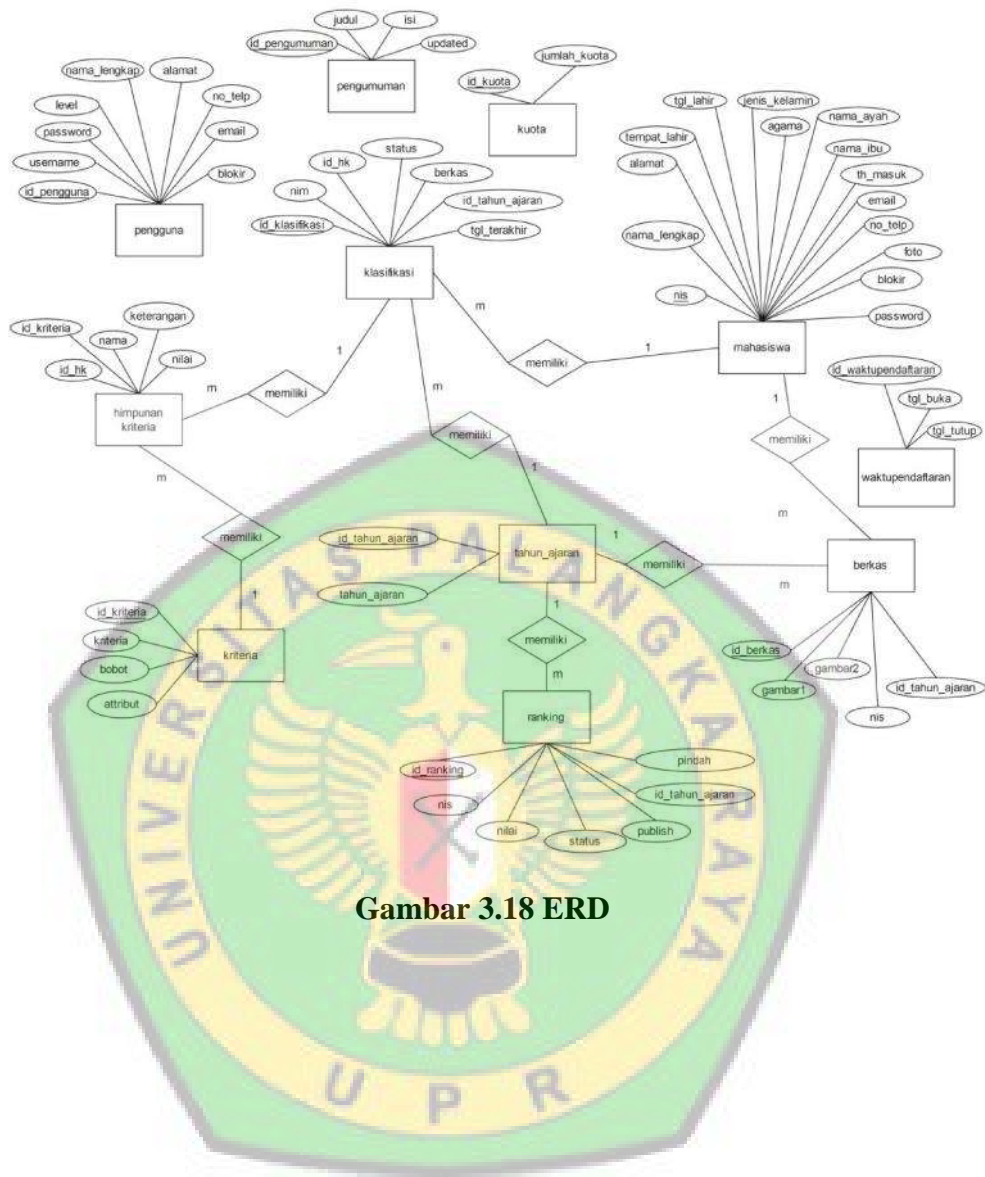


Gambar 3.17. DFD Level 4 Proses 3.3.2

3.3.1.3 *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) digunakan dalam membangun basis data untuk membangun hubungan dari dua atau lebih entitas/tabel. ERD terdiri dari 2 komponen utama yaitu entitas dan relasi. Kedua komponen tersebut diprediksikan lebih jauh melalui atribut atau properti. Hubungan antara entitas yang terjadi didalam sistem dapat dilihat pada gambar 3.18.





Gambar 3.18 ERD

3.3.2 Desain

3.3.2.1 Desain Tabel

Perancangan Basis Data merupakan proses pembentukan Basis Data yang merupakan tahapan yang sangat menentukan bagi terciptanya sistem yang baik. Perancangan Basis Data dalam Aplikasi Penjurusan Siswa SMAN 1 Kapuas Hilir berbasis *Web* ditunjukkan agar dalam pengoperasian dan pengimplementasiannya, dapat dilakukan pengolahan data yang maksimal.

Perancangan basis data Aplikasi Penjurusan Siswa SMAN 1 Kapuas Hilir berbasis *Web* ini memiliki 11 (sebelas) tabel, yaitu sebagai.

1. Perancangan Tabel pengguna

Tabel pengguna digunakan untuk menyimpan data Wakasek Bidang Kesiswaan dan data pimpinan (Kepala Sekolah) yang mengelola Aplikasi Penjurusan Siswa SMAN 1 Kapuas Hilir berbasis *web*. Tabel 3.10 berikut adalah struktur tabel pengguna.

Tabel 3.10. Tabel pengguna

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_pengguna	int	5	Primary key
2	username	varchar	20	Not Null
3	password	varchar	50	Not Null
4	level	varchar	100	Not Null
5	nama_lengkap	varchar	50	Not Null
6	alamat	varchar	100	Not Null
7	no_telp	int	20	Not Null
8	email	varchar	50	Not Null
9	blokir	enum	Y, N	Not Null

2. Perancangan Tabel siswa

Tabel siswa digunakan untuk menyimpan data siswa. Tabel 3.11 berikut adalah struktur tabel siswa.

Tabel 3.11. Tabel siswa

No	Nama Field	Tipe Data	Lebar	Keterangan
1	nis	varchar	20	Primary Key
2	nama_lengkap	varchar	100	Not null
3	password	varchar	50	Not Null
4	alamat	varchar	100	Not Null
5	tempat_lahir	varchar	100	Not Null
6	tgl_lahir	date		Not Null
7	jenis_kelamin	enum	L, P	Not Null
8	agama	varchar	20	Not Null
9	nama_ayah	varchar	100	Not Null
10	nama_ibu	varchar	100	Not Null
11	th_masuk	int	4	Not Null
12	email	varchar	50	Not Null
13	no_telp	int	20	Not Null
14	foto	varchar	150	Not Null
15	blokir	enum	Y, N	Not Null

3. Perancangan Tabel himpunankriteria

Tabel himpunankriteria digunakan untuk menyimpan data himpunan kriteria untuk menentukan penilaian penjurusan. Tabel 3.12 berikut adalah struktur tabel himpunankriteria.

Tabel 3.12. Tabel himpunankriteria

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_hk	int	5	Primary key
2	id_kriteria	int	5	Foreign Key
3	nama	varchar	60	Not Null
4	keterangan	varchar	15	Not Null
5	nilai	int	10	Not Null

4. Perancangan Tabel klasifikasi

Tabel klasifikasi berfungsi untuk untuk tempat menyimpan data klasifikasi tiap siswa. Tabel 3.13 berikut adalah struktur tabel klasifikasi.

Tabel 3.13. Tabel klasifikasi

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_klasifikasi	int	5	Primary key
2	nis	int	20	Foreign Key
3	id_hk	int	5	Foreign Key
4	status	varchar	10	Not Null
5	berkas	varchar	5	Not Null
6	id_tahun_ajaran	varchar	5	Foreign Key
7	tgl_terakhir	timestamp	-	current_timestamp

5. Perancangan Tabel kriteria

Tabel kriteria berfungsi untuk menyimpan data bobot kriteria dari masing-masing kriteria. Tabel 3.14 berikut adalah struktur tabel kriteria.

Tabel 3.14. Tabel kriteria

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_kriteria	int	5	Primary key
2	kriteria	varchar	100	Not Null
3	bobot	int	10	Not Null
4	attribut	varchar	10	Not Null

6. Perancangan Tabel kuota

Tabel kuota digunakan untuk menyimpan data kuota dalam penjurusan siswa. Tabel 3.15 berikut adalah struktur tabel kuota.

Tabel 3.15. Tabel kuota

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_kuota	int	5	Primary key
2	jumlah_kuota	int	10	Not Null

7. Perancangan Tabel pengumuman

Tabel pengumuman digunakan untuk menyimpan data pengumuman yang dikelola oleh admin. Tabel 3.16 berikut adalah struktur tabel pengumuman.

Tabel 3.16. Tabel pengumuman

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_pengumuman	int	5	Primary key
2	judul	varchar	100	Not Null
3	isi	text	-	Not Null
4	updated	timestamp	-	current_timestamp

8. Perancangan Tabel berkas

Tabel berkas digunakan untuk menyimpan berkas pendaftaran penjurusan siswa. Tabel 3.17 berikut adalah struktur tabel berkas.

Tabel 3.17. Tabel berkas

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_berkas	int	5	Primary key
2	gambar1	varchar	100	Not Null
3	gambar2	varchar	100	Not Null
5	nis	varchar	20	Foreign Key
6	id_tahun_ajaran	varchar	5	Foreign Key

9. Perancangan Tabel waktupendaftaran

Tabel waktupendaftaran digunakan untuk tempat menyimpan pengaturan waktu pendaftaran dan penutupan penjurusan siswa. Tabel 3.18 berikut adalah struktur tabel waktupendaftaran.

Tabel 3.18. Tabel waktupendaftaran

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_waktupendaftaran	int	5	Primary key
2	tgl_buka	date	-	Not Null
3	tgl_tutup	date	-	Not Null

10. Perancangan Tabel ranking

Tabel ranking digunakan untuk tempat menyimpan hasil perhitungan yang sudah terurut dan sesuai dengan jumlah kuota. Tabel 3.19 berikut adalah struktur tabel ranking.

Tabel 3.19. Tabel ranking

No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_ranking	int	5	Primary key
2	nis	varchar	20	Foreign Key
3	nilai	double	-	Not Null
4	status	enum	'Proses', 'Tidak Diproses'	Not Null
5	publish	enum	'Umumkan', 'Tidak Umumkan'	Not Null
6	id_tahun_ajaran	varchar	5	Foreign Key
7	pindah	varchar	5	Not Null
8	tanggal	datetime	-	Not Null

11. Perancangan Tabel tahun_ajaran

Tabel tahun_ajaran digunakan untuk tempat menampung data tahun ajaran. Tabel 3.20 berikut adalah struktur tabel tahun_ajaran.

Tabel 3.20. Tabel tahun_ajaran

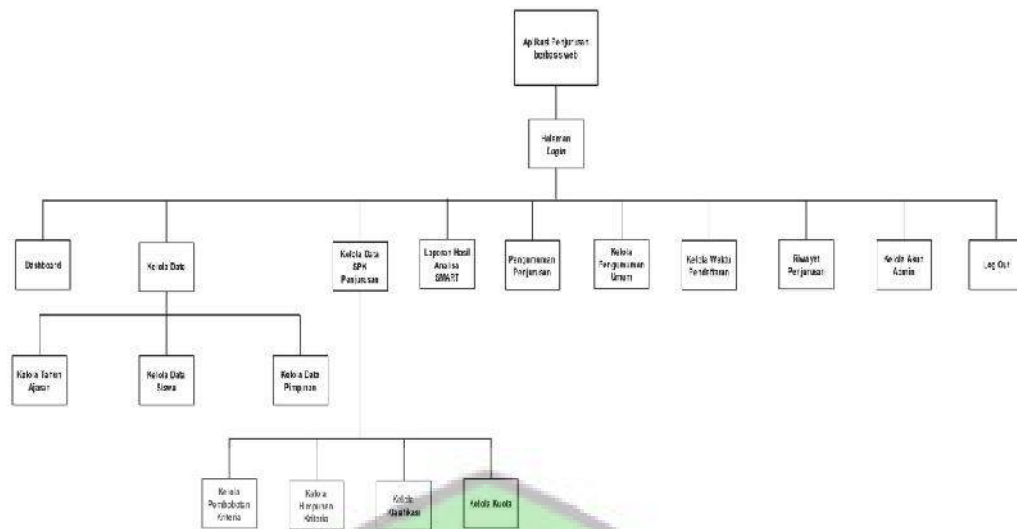
No	Nama Field	Tipe Data	Lebar	Keterangan
1	id_tahun_ajaran	int	5	Primary Key
2	tahun_ajaran	varchar	20	Not Null

3.3.2.2 Desain Navigasi

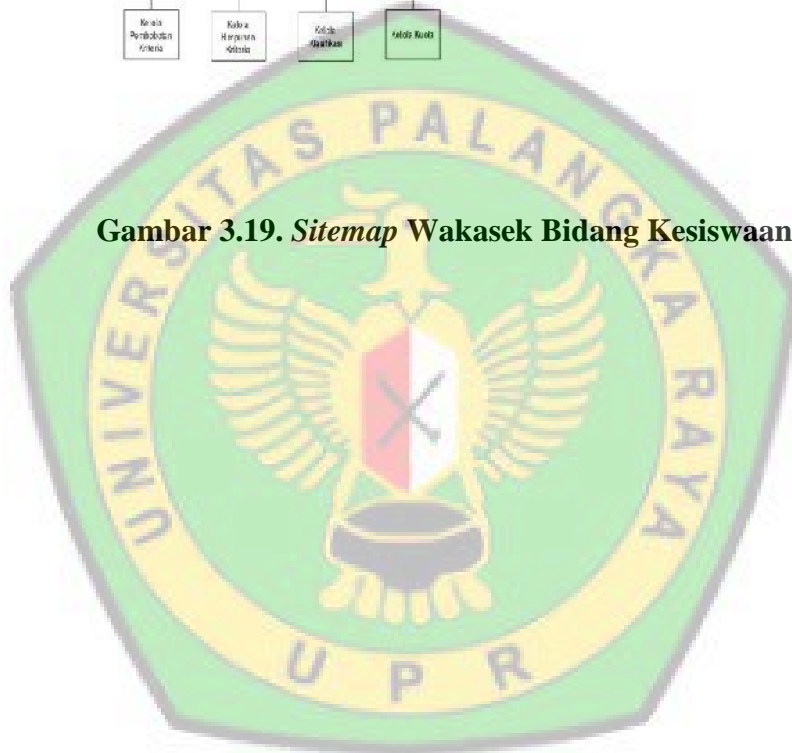
Desain navigasi menggunakan *Sitemap*, pada Sistem Aplikasi Penjurusan berbasis *website* ini antara lain *Sitemap* Wakasek Bidang Kesiswaan, *Sitemap* Siswa dan *Sitemap* Kepala Sekolah.

1. *Sitemap* Wakasek Bidang Kesiswaan

Sitemap Wakasek Bidang Kesiswaan pada aplikasi penjurusan di SMA Negeri 1 Kapuas Hilir berbasis *web* ini, yaitu seperti pada Gambar 3.19.



Gambar 3.19. Sitemap Wakasek Bidang Kesiswaan



2. Sitemap Siswa

Sitemap Siswa pada aplikasi penjurusan di SMA Negeri 1 Kapuas Hilir berbasis *web* ini, yaitu seperti pada Gambar 3.20.



Gambar 3.20. Sitemap Siswa

3. Sitemap Kepala Sekolah

Sitemap Kepala Sekolah pada aplikasi penjurusan di SMA Negeri 1 Kapuas Hilir berbasis *web* ini, yaitu seperti pada Gambar 3.21.



Gambar 3.21. Sitemap Kepala Sekolah

3.3.2.3 Desain Antarmuka Wakasek Bidang Kesiswaan

1. Halaman Login

Gambar 3.22 menggambarkan halaman login yang merupakan halaman pertama yang diakses oleh Wakasek Bidang Kesiswaan sebagai proses awal dalam melakukan pengelolaan. Untuk melakukan login, Wakasek Bidang Kesiswaan harus memasukkan *username* dan *password*.

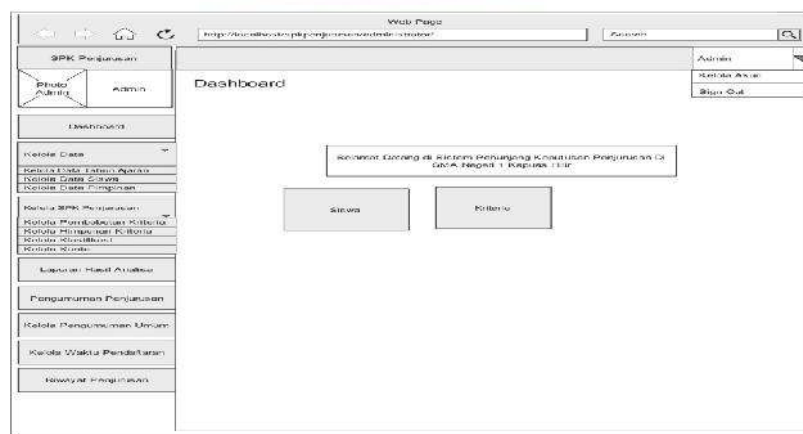


The screenshot shows a web interface for an administrator login. At the top left is a circular logo with the letter 'R' and the word 'LOGO' below it. To the right of the logo is the text 'Sekolah Menengah Atas Negeri 1 Kapuas Hilir'. Below this is a 'LOGIN ADMIN' section containing three input fields: 'Admin', 'Username', and 'Password'. A 'Sign in' button is located at the bottom right of the login form.

Gambar 3.22. Halaman Login Wakasek Bidang Kesiswaan

2. Halaman Dashboard

Gambar 3.23 menggambarkan halaman dashboard dari Wakasek Bidang Kesiswaan ketika telah berhasil login. Dalam halaman ini menampilkan *header* dan *sidebar*. Pada bagian *sidebar* merupakan fitur yang dapat diakses oleh Wakasek Bidang Kesiswaan, yaitu *Dashboard*, kelola data, kelola data SPK Penjurusan, Laporan Hasil Analisa, Pengumuman Penjurusan, Kelola Pengumuman Umum, Kelola Waktu Pendaftaran, Melihat Riwayat Penjurusan dan kelola akun.



Gambar 3.23. Halaman Dashboard Wakasek Bidang Kesiswaan

3. Halaman Kelola Tahun Ajaran

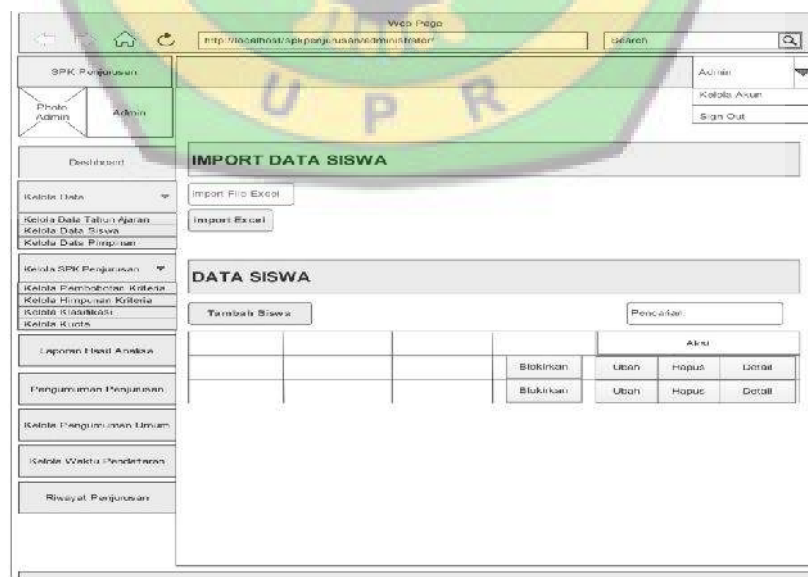
Gambar 3.24 merupakan halaman kelola tahun Ajaran dimana terdapat *button* tambah dan ubah tahun Ajaran.



Gambar 3.24. Halaman Kelola Tahun Ajaran

4. Halaman Kelola Data Siswa

Gambar 3.25 merupakan halaman kelola data Siswa dimana pada halaman ini menampilkan data-data Siswa yang telah diinputkan yang ditampung dalam *database* dan ditampilkan dalam bentuk tabel. Pada halaman ini terdapat aksi tambah, ubah, impor, blokirkan, buka blokir, detail, cari dan hapus data Siswa.



Gambar 3.25. Halaman Kelola Data Siswa

5. Halaman Kelola Data Pimpinan

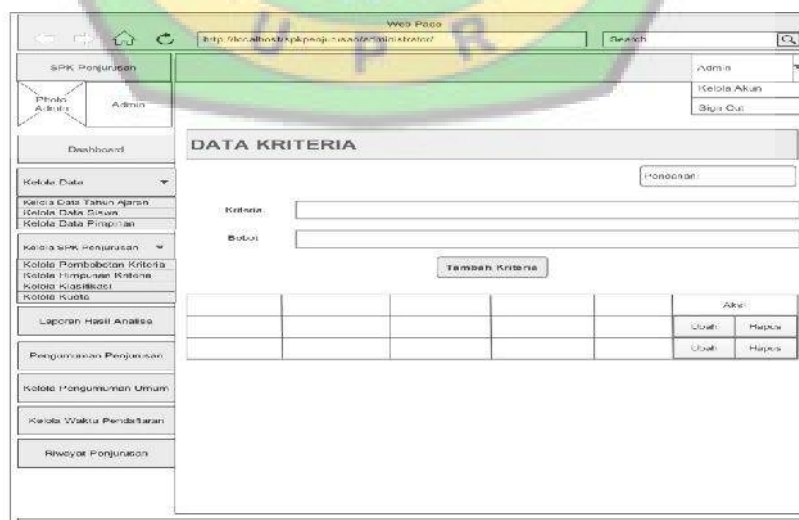
Gambar 3.26 merupakan halaman kelola data Pimpinan dimana pada halaman ini menampilkan data pimpinan yang mengelola penjurusan siswa. Pada halaman ini terdapat aksi ubah data pimpinan.



Gambar 3.26. Halaman Kelola Data Pimpinan

6. Halaman Kelola Pembobotan Kriteria

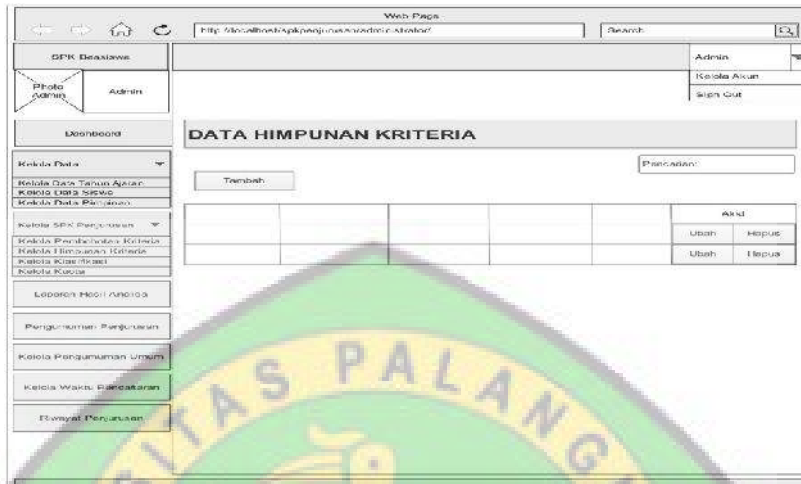
Gambar 3.27 merupakan halaman kelola pembobotan kriteria dimana pada halaman ini menampilkan data-data kriteria beserta bobot yang telah diinputkan, yang ditampung dalam *database* dan ditampilkan dalam bentuk tabel. Pada halaman ini terdapat aksi tambah, ubah dan hapus data kriteria.



Gambar 3.27. Halaman Kelola Pembobotan Kriteria

7. Halaman Kelola Himpunan Kriteria

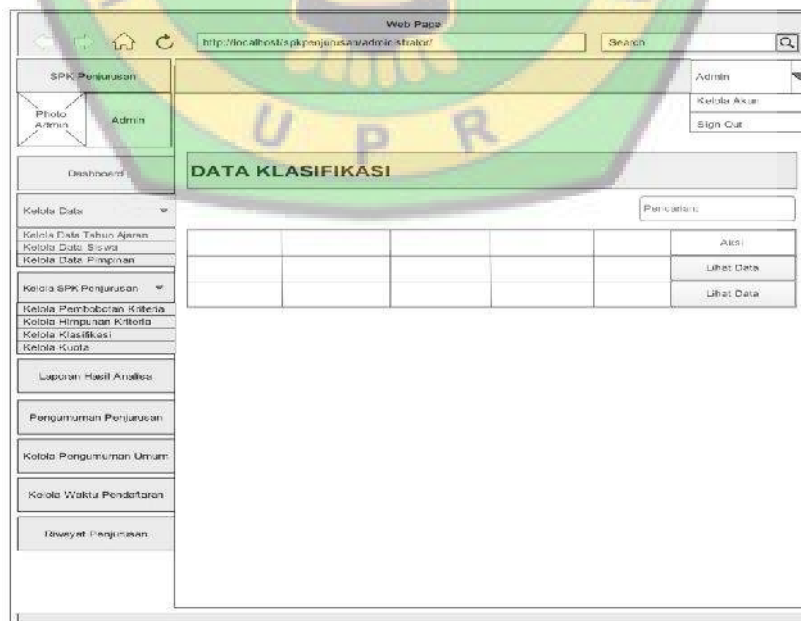
Gambar 3.28 merupakan halamankelola himpunan kriteria. Pada halaman ini terdapat *button* input himpunan kriteria yang akan membawa pada halaman yang mengelola himpunan kriteria seperti tambah, ubah dan hapus himpunan kriteria.



Gambar 3.28. Halaman Kelola Himpunan Kriteria

8. Halaman Kelola Klasifikasi

Gambar 3.29 merupakan halaman kelola klasifikasi. Pada halaman ini terdapat *button* lihat data klasifikasi dan berkas yang diinputkan oleh Siswa.



Gambar 3.29. Halaman Kelola Klasifikasi

9. Halaman Kelola Kuota

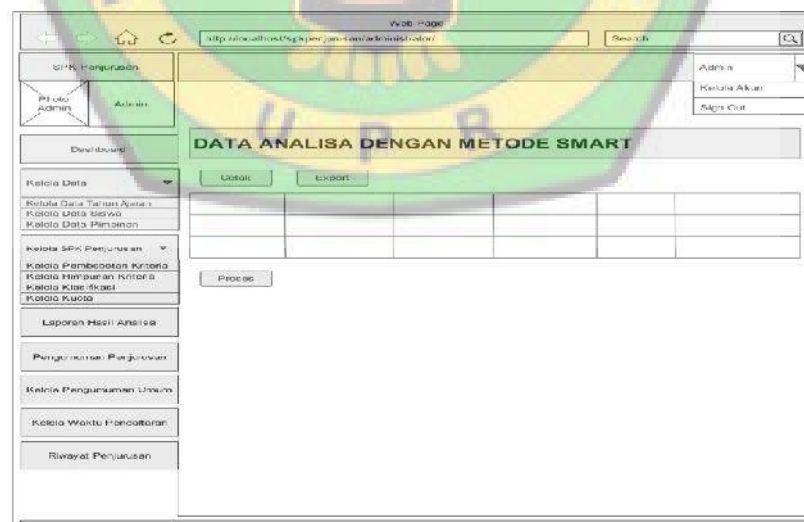
Gambar 3.30 merupakan halaman kelola kuota. Pada halaman ini terdapat *button* ubah untuk mengubah kuota penjurusan.



Gambar 3.30. Halaman Kelola Kuota

10. Laporan Hasil Analisa

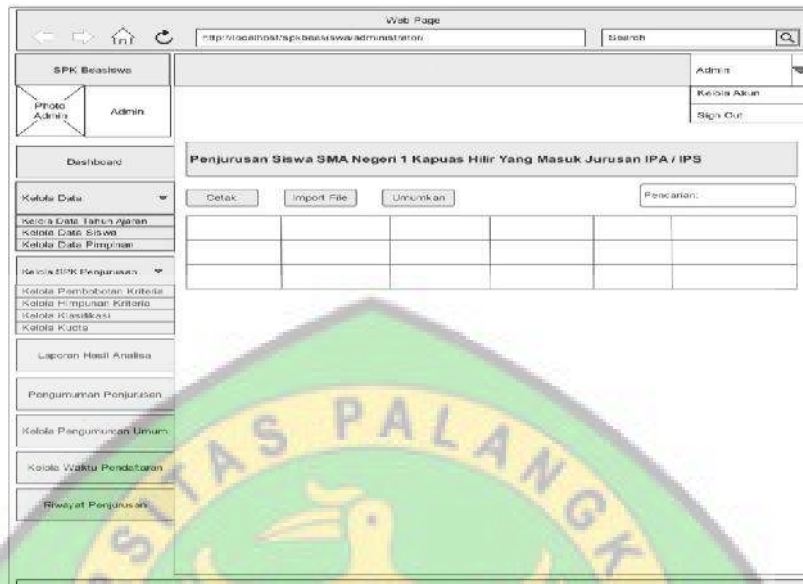
Gambar 3.31 merupakan halaman yang menampilkan hasil analisa dari perhitungan SMART.



Gambar 3.31. Laporan Hasil Analisa

11. Halaman Pengumuman Penjurusan

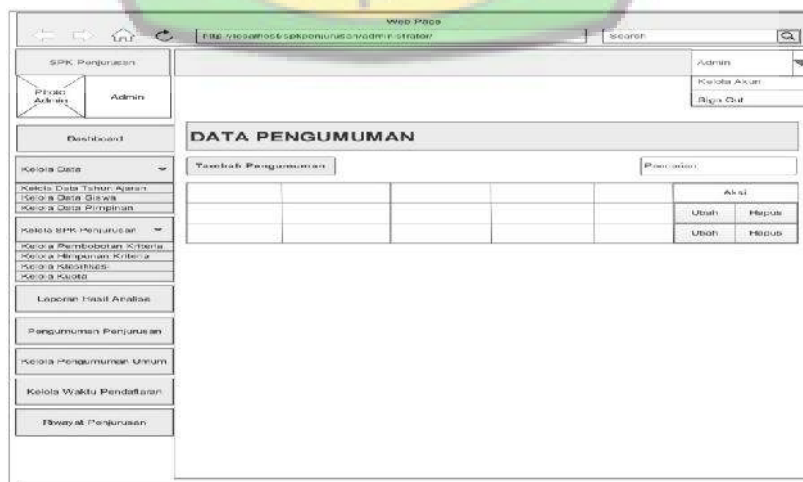
Gambar 3.32 merupakan halaman pengumuman penjurusan dimana pada halaman ini menampilkan pengumuman penjurusan siswa. Pada halaman ini terdapat aksi cetak, export file dan umumkan/tidak umumkan.



Gambar 3.32. Halaman Pengumuman Penjurusan

12. Halaman Kelola Pengumuman Umum

Gambar 3.33 merupakan halaman kelola pengumuman umum dimana pada halaman ini menampilkan data pengumuman yang telah diinputkan, yang ditampung dalam *database* dan ditampilkan dalam bentuk tabel. Pada halaman ini terdapat aksi tambah, ubah dan hapus pengumuman.



Gambar 3.33. Halaman Kelola Pengumuman Umum

13. Halaman Kelola Waktu Pendaftaran

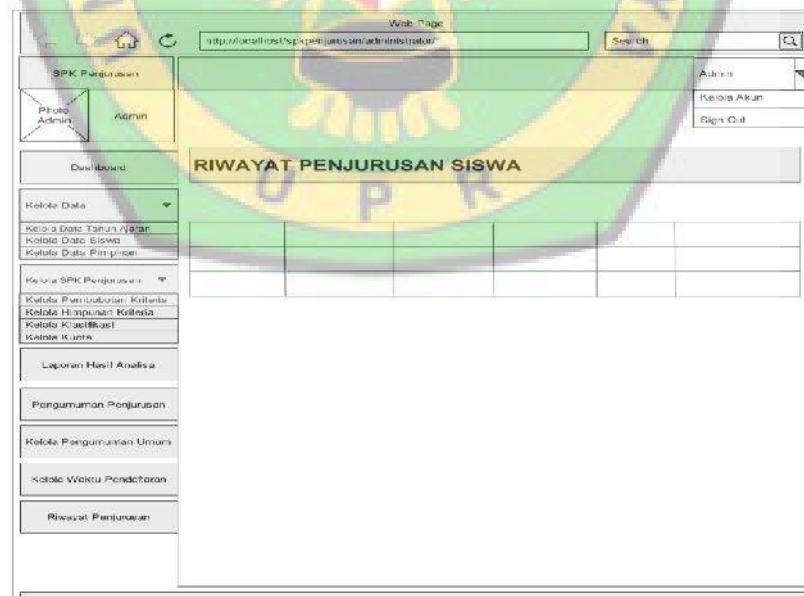
Gambar 3.34 merupakan halaman kelola waktu pendaftaran. Pada halaman ini terdapat *button* ubah untuk mengubah jadwal pendaftaran penjurusan siswa.



Gambar 3.34. Halaman Waktu Pendaftaran

14. Halaman Riwayat Penjurusan

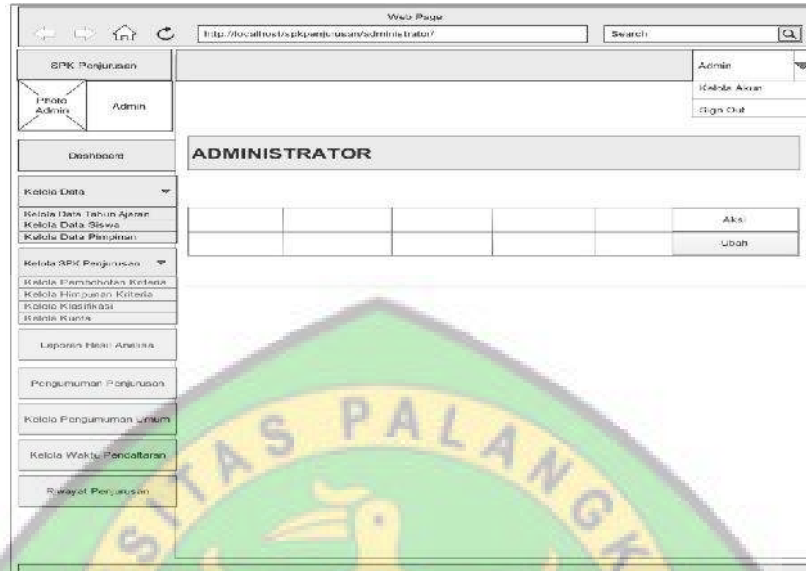
Gambar 3.35 merupakan halaman untuk melihat riwayat penjurusan siswa berdasarkan tahun Ajaran.



Gambar 3.35. Halaman Riwayat Penjurusan

15. Halaman Kelola Akun Wakasek Bidang Kesiswaan

Gambar 3.36 merupakan halaman kelola akun Wakasek Bidang Kesiswaan. Pada halaman ini terdapat *button* ubah untuk mengubah data Wakasek Bidang Kesiswaan.



Gambar 3.36. Halaman Kelola Akun Wakasek Bidang Kesiswaan

3.3.2.4 Desain Antarmuka Siswa

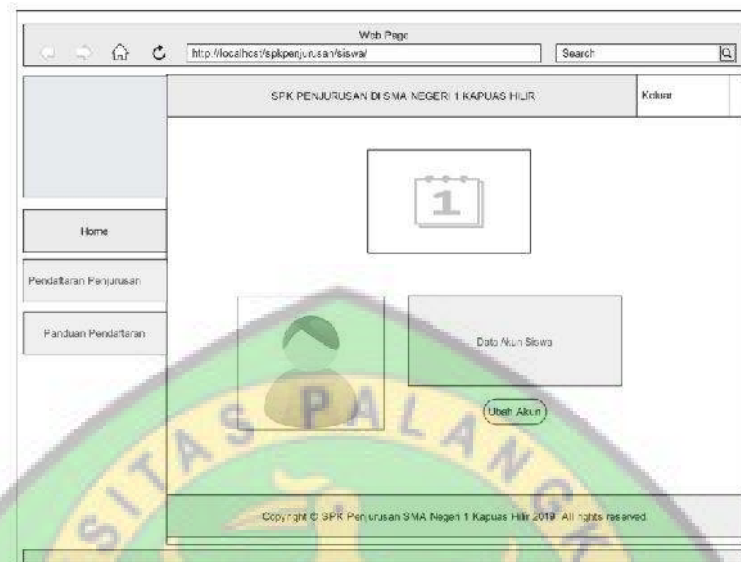
1. Halaman Login

Gambar 3.37 menggambarkan halaman login yang merupakan halaman pertama yang diakses oleh Siswa sebagai proses awal ketika ingin masuk ke sistem. Untuk melakukan login, Siswa harus memasukkan nis dan *password*.

Gambar 3.37. Halaman Login Siswa

2. Halaman Home

Gambar 3.38 menggambarkan halaman home dari Siswa ketika telah berhasil login. Dalam halaman ini menampilkan bagian menu *website* yaitu Home, Daftar Penjurusan, Panduan dan Kelola Akun.



Gambar 3.38. Halaman Home

3. Halaman Form Pendaftaran

Gambar 3.39 menggambarkan halaman pendaftaran penjurusan yang diisi oleh Siswa.

Gambar 3.39. Halaman Pendaftaran

4. Halaman Pengumuman

Gambar 3.40 menggambarkan halaman pengumuman hasil penjurusan siswa.



Gambar 3.40. Halaman Pengumuman

5. Halaman Panduan Pendaftaran

Gambar 3.41 menggambarkan halaman panduan pendaftaran penjurusan di *website* Sistem Penunjang Keputusan Penjurusan Siswa.

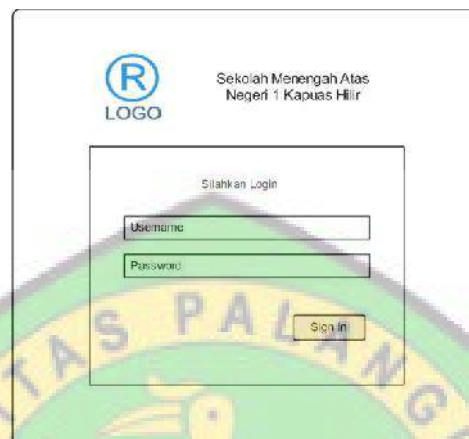


Gambar 3.41. Halaman Panduan Pendaftaran

3.3.2.5 Desain Antarmuka Kepala Sekolah

1. Halaman Login

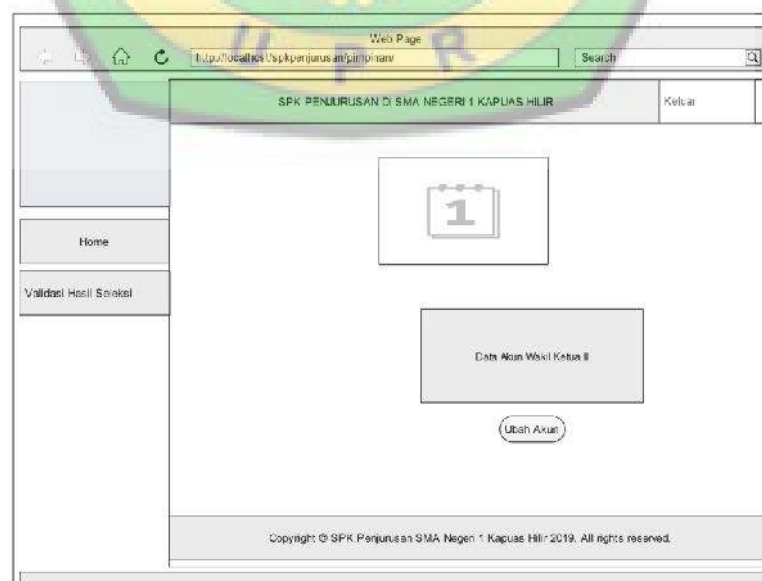
Gambar 3.42 menggambarkan halaman login yang merupakan halaman pertama yang diakses oleh Kepala Sekolah sebagai proses awal ketika ingin masuk ke sistem. Untuk melakukan login, kepala sekolah harus memasukkan *username* dan *password*.



Gambar 3.42. Halaman Login Kepala Sekolah

2. Halaman Home

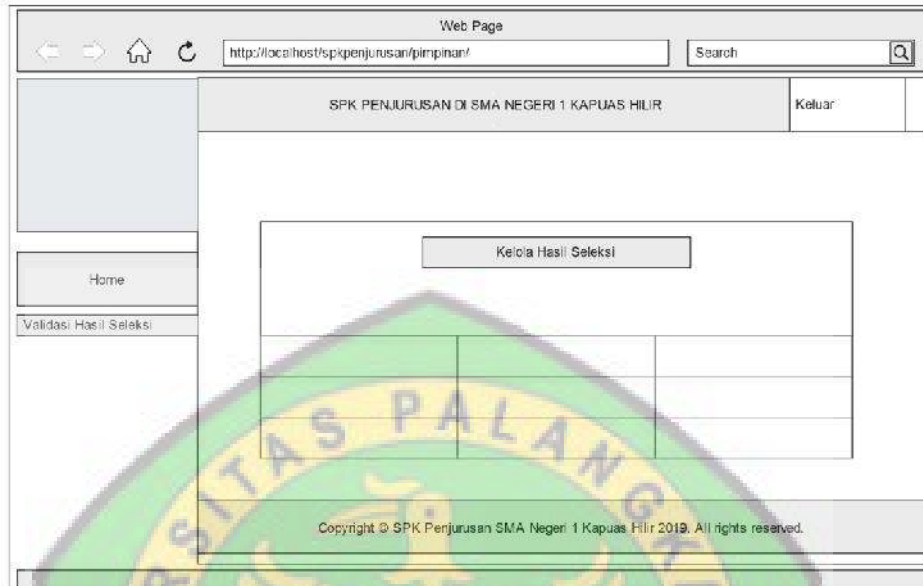
Gambar 3.43 menggambarkan halaman home ketika telah berhasil login. Dalam halaman ini menampilkan bagian menu Validasi Hasil Seleksi.



Gambar 3.43. Halaman Home

3. Halaman Validasi Hasil Seleksi

Gambar 3.44 menggambarkan halaman untuk validasi hasil seleksi yang sebelumnya menggunakan metode SMART. Kepala sekolah memutuskan apakah memproses siswa tersebut atau tidak.



Gambar 3.44. Halaman Validasi Hasil Seleksi

BAB V PENUTUP

5.1. Kesimpulan

Kesimpulan yang didapatkan dari penyusunan Tugas Akhir yang berjudul Rancang Bangun Aplikasi Penjurusan di SMAN 1 Kapuas Hilir Menggunakan Metode *Simple Multi Attribute Rating Technique* (SMART) Berbasis *WEB* yaitu sistem ini dibuat dengan menggunakan metode pengembangan perangkat lunak *Waterfall* dengan tahapan *Communication* (Komunikasi), *Planning* (Perencanaan), *Modeling* (Permodelan), *Construction* (Konstruksi), *Deployment* (Penyerahan). Penelitian diawali dengan mengumpulkan dan mempelajari literatur yang berkaitan dengan teori sistem pendukung keputusan menggunakan metode *Simple Multi Attribute Rating Technique* (SMART) yang berbasis *web*. Tahapan metode SMART yaitu menentukan jumlah kriteria; melakukan normalisasi; memberikan nilai kriteria untuk setiap alternatif; menghitung nilai utility; dan menghitung nilai akhir masing-masing.

Selanjutnya pengumpulan data yang dibagi menjadi dua, yaitu data primer yang diperoleh dari objek penelitian di lapangan dan data sekunder yang diperoleh secara tidak langsung dari narasumber. Perancangan yang dimulai dari perancangan basis data, lalu perancangan sistem dan terakhir perancangan antarmuka. Kemudian sistem diimplementasikan dalam bentuk program berbasis *web* menggunakan bahasa pemrograman PHP dan MySQL sebagai *databasenya*. Proses terakhir pengujian yang dilakukan antarmuka terhadap pengguna dan program, yang dilakukan menggunakan *blackbox testing* sebagai uji coba sistem dan penyebaran kuesioner sebagai uji coba terhadap pengguna yang kemudian dilakukan perhitungan menggunakan skala likert. Maka telah dibangun sebuah Aplikasi Penjurusan di SMAN 1 Kapuas Hilir Menggunakan Metode *Simple Multi Attribute Rating Technique* (SMART) Berbasis *WEB* yang dapat menjangkau penjurusan tiap siswa berdasarkan kriteria-kriteria yang ditentukan.

Berdasarkan uji coba terhadap pengguna melalui kuesioner, disimpulkan bahwa Aplikasi Penjurusan di SMAN 1 Kapuas Hilir ini sangat layak

digunakan dengan nilai akhir rata-rata 33 (dari nilai 35 tertinggi) dan dengan presentase rata-rata 94,28% yang didapatkan melalui perhitungan skala likert. Dengan adanya aplikasi penjurusan ini, dapat memberikan alternatif solusi dalam penjurusan siswa di SMA Negeri 1 Kapuas Hilir.

5.2. Saran

Adapun saran yang bisa diberikan untuk pengembangan sistem ini yaitu selain dalam bentuk *WEB*, Aplikasi Penjurusan di SMAN 1 Kapuas Hilir Berbasis *WEB* dapat dibuat versi *mobile* sehingga memudahkan siswa untuk mengupload file berkas dengan cara mengambil foto langsung menggunakan hp tanpa harus scan menggunakan printer dan komputer/laptop.



DAFTAR PUSTAKA

- Bukukun. 2014. *Pengertian Sekolah*. (<https://id.scribd.com/document/317117727/Pengertian-Sekolah>). Diakses pada 26 Februari 2019.
- Cahya. 2018. *Simple Multi Attribute Rating Technique (SMART)*. (<https://cahyadsn.phpindonesia.id/extra/smart.php>). Diakses pada 26 Februari 2019.
- Dinas Pendidikan Provinsi Kalimantan Tengah. 2018. *Pedoman Pelaksanaan Penerimaan Peserta Didik Baru (PPDB) Provinsi Kalimantan Tengah Tahun Pelajaran 2018/2019*. Dinas Pendidikan Provinsi Kalimantan Tengah: Kalimantan Tengah.
- El Said, Fairuz. 2010. *Sistem Basis Data – Konsep Basis Data*. (<https://fairuzelsaid.com/2010/01/22/sistem-basis-data-konsep-basis-data/>). Diakses pada 30 Maret 2019.
- Fathansyah. 2007. *Basis Data*. Bandung: Informatika.
- Haniif. 2007. *Sistem Pendukung Keputusan (SPK)*. (<https://haniif.com/tag/sistem-pendukung-keputusan/>). Diakses pada 30 Maret 2019.
- Harsiti. 2012. *Sistem Basis Data*. (<https://www.academia.edu/5528409/Sistembasis-data>). Diakses pada 30 Maret 2019.
- Kadir, Abdul. 2003. *Pemrograman Web Mencakup : HTML, CSS, Javascript dan PHP*. Yogyakarta: Penerbit Andi Offset.
- Kusnadi, Engkus. 2013. *Pengertian dan Fungsi HTML (HyperText Markup Language)*. (<http://mypctutorel.blogspot.co.id/2013/11/pengertian-dan-fungsi-html-hypertext.html>). Diakses pada 30 Maret 2019.
- Lutfi Indrawan, Muhammad. 2013. *Pengenal dan Instalasi XAMPP*. Bandung: Institut Teknologi Bandung.
- McLeod. Raymond. 1998. *Management Information Systems*. 7th Edition, New Jersey: Prentice Hall, Inc.

- Mufizar, Teuku. 2016. *Sistem Pendukung Keputusan Pemilihan Jurusan dengan menggunakan Metode SAW Di SMA 6 Tasikmalaya*. Jurnal VOI STMIK Tasikmalaya Vol.5.
- Purwanti, Dian. 2008. *Pengertian Website*. (<http://deeyaan.com/2008/03/pengertian-website.html>). Diakses pada 30 Maret 2019.
- Rahman, Syakiru. 2013. *Apa itu CMS (Content Management System)*. (<https://www.devaradise.com/id/2013/11/mengenal-pengertian-cms-content-management-system.html>). Diakses pada 30 Maret 2019.
- Rosa A. Sdan M. Shalahuddin. 2011. *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Modula: Bandung.
- Subakti, Irfan. 2002. *Sistem Pendukung Keputusan (Decision Support System)*. Yogyakarta: Graha Ilmu.
- Syafrizal, Melwin. 2010. *Sistem Pendukung Keputusan (Decision Support System)*. (<https://media.neliti.com/.../90157-ID-sistem-pendukung-keputusan-decisin-suppo.pdf>). Diakses pada 30 Maret 2019.
- Turban, Efraim; Aronson, Jay E. Dan Liang, Ting-Peng. 2005. *Decision Support Systems and Intelligent Systems*. Yogyakarta: Andi.
- Utami, Ema; Sukrisno. 2005. *10 Langkah Belajar Logika dan Algoritma. menggunakan Bahasa C dan C++ di GNU/Linux*. STMIK Amikom Yogyakarta: Andi.