



**RANCANG BANGUN SISTEM PENJADWALAN UJIAN BERBASIS WEB
PADA SMK 1 TROWULAN DENGAN METODE ALGORITMA
GENETIKA**

TUGAS AKHIR

Program Studi

S1 Sistem Informasi

Oleh :

ARIEF NUR ABDULLAH

16410100076

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA
2023**

**RANCANG BANGUN SISTEM PENJADWALAN UJIAN BERBASIS WEB
PADA SMK 1 TROWULAN
DENGAN METODE ALGORITMA GENETIKA**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana Komputer**



Oleh :
Nama : Arief Nur Abdullah
NIM : 16410100076
Program : S1 (Strata Satu)
Jurusan : Sistem Informasi

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA**

2023

TUGAS AKHIR

RANCANG BANGUN SISTEM PENJADWALAN UJIAN BERBASIS WEB PADA SMK 1 TROWULAN DENGAN METODE ALGORITMA GENETIKA

Dipersiapkan dan disusun oleh

Arief Nur Abdullah

NIM: 16410100076

Telah diperiksa, diuji dan disetujui oleh Dewan Pembahas

Pada:

Susunan Dewan Pembahas


Pembimbing:


- I. Slamet, M.T.
NIDN. 0701127503
- II. Erwin Sutomo, S.Kom., M.Eng.
NIDN. 0722057501

Pembahas :

- I. Dr. Eng. Valentinus Roby Hananto, S.Kom., M.Sc.
NIDN: 0715028903


Digitally signed by Slamet A.
DN: cn=Slamet A, o=Universitas
Dinamika, ou=Prodi S1 Sistem
Informasi,
email=slamet@dinamika.ac.id,
c=ID
Date: 2023.08.21 10:42:01 +07'00'


2023.08.21
11:03:48
+07'00'



Tugas Akhir ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana



Digitally signed by
Universitas Dinamika
Date: 2023.08.22
08:00:42 +07'00'

Tri Sagirani, S.Kom., M.MT

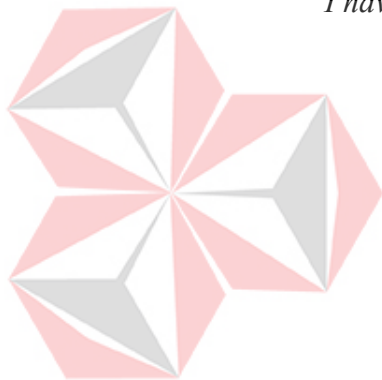
NIDN: 0731017601

Dekan Fakultas Teknologi dan Informatika

UNIVERSITAS DINAMIKA

“I have not failed. I’ve just found 10,000 ways that won’t work”

- Thomas A, Edison -



UNIVERSITAS
Dinamika

PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa **Universitas Dinamika**, saya:

Nama : Arief Nur Abdullah
NIM : 16410100076
Program Studi : S1 Sistem Informasi
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Tugas Akhir
Judul Karya : **RANCANG BANGUN SISTEM PENJADWALAN
UJIAN BERBASIS WEB PADA SMK 1 TROWULAN
DENGAN METODE ALGORITMA GENETIKA**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Universitas Dinamika Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar keserjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 04 Agustus 2023

Yang menyatakan



Arief Nur Abdullah
NIM: 16410100076

ABSTRAK

SMK Negeri 1 Trowulan adalah salah satu sekolah menengah kejuruan yang terletak di daerah kabupaten Mojokerto. Proses pembuatan penjadwalan dimulai dari proses melihat daftar kelas, daftar mata pelajaran, dan daftar jurusan. Setelah itu melakukan pengecekan ketersediaan ruangan dan jumlah kelas yang mengikuti ujian tersebut. Jika sudah selesai Bagian kurikulum membuat penjadwalan. Bagian Kurikulum kesulitan dalam proses pengecekan ketersediaan ruangan yang digunakan untuk ujian yang dijadwalkan. Selain itu Bagian Kurikulum juga kesulitan dalam menjadwalkan pengawas ujian serta kelas yang mengikuti ujian karena pengawas ujian memiliki beban jaga ujian yang berbeda antara Guru PNS dan Non PNS dan kelas yang mengikuti ujian karena keterbatasan setiap mata pelajaran dibagi menjadi dua sesi dan satu sesinya diikuti oleh 5 kelas saja. Oleh karena itu Bagian Kurikulum harus menyusun ulang kembali dengan cara pengecekan kapasitas ruangan dan jumlah kelas yang mengikuti ujian untuk dipindahkan agar jumlah kapasitas ruangan terpenuhi dan menyesuaikan beban jaga pengawas dengan jadwal ujian yang sudah dijadwalkan dengan menggunakan Microsoft Excel. Maka solusi yang dibutuhkan adalah sistem penjadwalan ujian dengan metode Algoritma Genetika. Hasil penelitian menunjukkan bahwa aplikasi penjadwalan sekolah ini telah menghasilkan jadwal yang ditampilkan setiap satu minggu sesuai dengan syarat dan ketentuan yang berlaku. Selama pengujian penjadwalan dengan melibatkan 3 individu, menggunakan probabilitas *CrossOver* sebesar 0.8, probabilitas mutasi sebesar 0.6, dan melakukan 100 generasi, penelitian ini berhasil menemukan jadwal yang memenuhi semua kriteria dan batasan yang telah ditetapkan. Pencapaian ini terjadi dalam 1 kali uji coba. Hasil yang memuaskan ini terlihat saat mencapai generasi ke-2 pada tahap mutasi, di mana sistem berhasil menghasilkan sebuah jadwal yang optimal sesuai dengan tujuan penelitian dan memenuhi semua kriteria serta batasan yang ada.

Kata Kunci : Penjadwalan, Algoritma Genetika, Probabilitas

KATA PENGANTAR

Puji Syukur kehadiran Allah Subhanahu Wa Ta'ala atas segala anugerah rahmat serta inayah-Nya, Penulis dapat menyelesaikan penelitian dan menyusun sebuah Laporan Tugas Akhir dengan judul “Rancang Bangun Sistem Penjadwalan Ujian Berbasis Web Pada SMK 1 Trowulan Dengan Metode Algoritma Genetika”. Laporan Tugas Akhir ini disusun dalam rangka penulisan laporan untuk persyaratan menyelesaikan Program Sarjana Komputer pada Program Studi S1 Sistem Informasi Universitas Dinamika.

Dalam melakukan penelitian maupun penyusunan laporan ini, penulis banyak mendapatkan dukungan dan dorongan dari berbagai pihak secara langsung maupun tidak langsung. Sehingga pada kesempatan ini, penulis menyampaikan ucapan terima kasih sebesar-besarnya kepada semua pihak yang telah membantu dalam menyelesaikan Laporan Tugas Akhir ini, terutama kepada:

1. Ibu dan Bapak tercinta yang selalu mendukung, mendoakan, dan memberikan semangat kepada Penulis untuk menyelesaikan Tugas Akhir ini.
2. Bapak Prof. Dr. Budi Jatmiko, M.Pd. selaku Rektor Universitas Dinamika Surabaya.
3. Ibu Tri Sagirani, S.Kom.,M.MT. selaku Dekan Fakultas Teknologi Informatika yang telah memberikan saran dan arahan, motivasi, dan dukungan dalam proses penyelesaian Tugas Akhir.
4. Bapak Dr. Anjik Sukmaaji, S.Kom., M.Eng. selaku Ketua Program Studi Sistem Informasi Universitas Dinamika Surabaya yang telah memberikan saran dan arahan, motivasi, dan dukungan dalam proses penyelesaian Tugas Akhir.
5. Bapak Slamet, M.T. selaku dosen pembimbing pertama yang telah memberikan saran dan arahan, motivasi, dan dukungan dalam proses penyelesaian Tugas Akhir.
6. Bapak Erwin Sutomo, S.Kom., M.Eng. selaku dosen pembimbing kedua yang telah memberikan saran dan arahan, motivasi, dan dukungan dalam proses penyelesaian Tugas Akhir.

7. Bapak Dr. Eng. Valentinus Roby Hananto, S.Kom., M.Sc. selaku dosen pembahas yang telah memberikan saran dan arahan, motivasi, dan dukungan dalam proses penyelesaian Tugas Akhir.

Semoga laporan Tugas Akhir ini dapat diterima dengan baik dan bermanfaat bagi penulis maupun semua pihak. Semoga Allah Subhanahu wa Ta'ala membalas semua kebaikan yang telah diberikan untuk membantu penulis.

Surabaya, 04 Agustus 2023

Arief Nur Abdullah



UNIVERSITAS
Dinamika

DAFTAR ISI

	Halaman
PERNYATAAN.....	i
ABSTRAK	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	v
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	xi
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	3
1.5. Manfaat.....	3
BAB II LANDASAN TEORI.....	4
2.1. Penelitian Terdahulu.....	4
2.2. Penjadwalan.....	5
2.3. <i>System Development Life Cycle</i>	5
2.4. Bagan Alir.....	7
2.5. Data Flow Diagram	8
2.6. Algoritma Genetika	8
BAB III METODOLOGI PENELITIAN.....	11
3.1. Communication	11
3.1.1. Project Initiation.....	11
3.1.2. Requirement Gathering	14
3.2. Planning.....	17
3.3. Modelling	17
3.3.1. Process Model	18
3.3.2. Data Model.....	19
3.3.3. Desain Sistem.....	24
BAB IV HASIL DAN PEMBAHASAN	25

4.1	Kebutuhan Sistem	25
4.1.1.	Spesifikasi Sistem	25
4.1.2.	Implementasi Sistem	26
4.1.3.	Hasil Uji Coba Sistem	28
4.2.	Evaluasi Sistem	44
BAB V KESIMPULAN DAN SARAN.....		49
5.1.	Kesimpulan.....	49
5.2.	Saran.....	49
DAFTAR PUSTAKA		49



UNIVERSITAS
Dinamika

DAFTAR GAMBAR

Halaman

Gambar 2.1. Tahapan <i>System Development Life Cycle</i> model <i>waterfall</i>	6
Gambar 3.1. Metodologi Penelitian	11
Gambar 3.2. Proses Bisnis	13
Gambar 3.3. IPO Diagram	18
Gambar 3.4. System Flow Diagram Lihat Jadwal	19
Gambar 3.5. Context Diagram	20
Gambar 3.6. Diagram Berjenjang	20
Gambar 3.7. Data Flow Diagram Level 0	21
Gambar 3.8. Data Flow Diagram Level 1 Data Master	22
Gambar 3.9. Conceptual Data Model.....	23
Gambar 3.10. Physical Data Model	23
Gambar 4.1. Tampilan Halaman Beranda.....	26
Gambar 4.2. Tampilan Halaman Awal Buat Jadwal (Untuk Ujian)	26
Gambar 4.3. Tampilan Halaman Berhasil Generate	27
Gambar 4.4. Tampilan Jadwal Pada User Siswa, Guru, Kepsek	27
Gambar 4.5. Uji Coba Jumlah Individu	29
Gambar 4.6. Uji Coba Probabilitas Crossover	30
Gambar 4.7. Uji Coba Probabilitas Mutasi	31
Gambar 4.8. Penentuan Parameter GA	32
Gambar 4.9. Request Waktu Satu	32
Gambar 4.10. Request Waktu Kedua	33
Gambar 4.11. Populasi Awal	33
Gambar 4.12. Hitung Nilai Fitness	37
Gambar 4.13. Proses Probabilitas Komulatif.....	38
Gambar 4.14. Proses Seleksi.....	39
Gambar 4.15. Crossover.....	41
Gambar 4.16. Mutasi Generasi Acak	42
Gambar 4.17. Hasil Mutasi	42
Gambar 4.18. Gen Acak Generasi 2.....	43

Gambar 4.19. Hasil Perolehan Mutasi Generasi 2	43
Gambar 4.20. Hasil Jadwal Ujian	44
Gambar 4.21. Contoh Crash Jadwal.....	45
Gambar 4.22. Waktu Yang Dibutuhkan.....	46
Gambar 4.23. Request Pengawas Pengganti	47
Gambar 4.24. Hasil Jadwal Ujian	48
Gambar 4.25. Hasil Jadwal Ujian	48
Gambar L6.1. Desain Data Interface Input Mata Pelajaran	68
Gambar L6.2. Desain Data Interface Output Mata Pelajaran	68
Gambar L6.3. Desain Data Interface Output Jurusan	68
Gambar L6.4. Desain Data Interface Output Jurusan	69
Gambar L6.5. Desain Output Request Waktu.....	69
Gambar L6.6. Desain Input Request Waktu	69
Gambar L6.7. Desain Output User.....	70
Gambar L6.8. Desain Input User	70
Gambar L6.9. Desain Output Data Guru	70
Gambar L6.10. Desain Input Data Guru	71
Gambar L6.11. Desain Output Data Ruang	71
Gambar L6.12. Desain Input Data Ruang.....	71
Gambar L6.13. Desain Output Data Request Waktu	72
Gambar L6.14. Desain Input Data Request Waktu.....	72
Gambar L6.15. Desain Input Data Penjadwalan.....	72
Gambar L6.16. Desain Output Data Penjadwalan	73
Gambar L8.1. Tampilan Halaman Awal Data Mata Pelajaran	78
Gambar L8.2. Tampilan Halaman Form Data Mata Pelajaran	78
Gambar L8.3. Tampilan Halaman Form Data Mata Pelajaran Harus Diisi	79
Gambar L8.4. Alert Data Mata Pelajaran Tersimpan	79
Gambar L8.5. Data Mata Pelajaran Tersimpan.....	79
Gambar L8.6. Alert Data Mata Pelajaran Diubah.....	80
Gambar L8.7. Tampilan Data Mata Pelajaran Berhasil Diubah	80
Gambar L8.8. Tampilan Halaman Data Mata Pelajaran Berhasil Dihapus	80
Gambar L8.9. Tampilan Halaman Awal Data Jurusan	81

Gambar L8.10. Tampilan Halaman Form Data Jurusan	81
Gambar L8.11. Tampilan Halaman Form Data Jurusan Harus Diisi.....	82
Gambar L8.12. Tampilan Halaman Data Jurusan Berhasil Disimpan.....	82
Gambar L8.13. Tampilan Halaman Data Jurusan Berhasil Diubah.....	83
Gambar L8.14. Tampilan Halaman Data Jurusan Berhasil Dihapus	83
Gambar L8.15. Tampilan Halaman Awal Data Guru	84
Gambar L8.16. Tampilan Halaman Form Data Guru	84
Gambar L8.17. Tampilan Halaman Form Data Guru Harus Diisi.....	85
Gambar L8.18. Tampilan Halaman Data Guru Berhasil Disimpan 1	85
Gambar L8.19. Tampilan Halaman Data Guru Berhasil Disimpan 2.....	85
Gambar L8.20. Tampilan Halaman Data Guru Berhasil Diubah 1.....	86
Gambar L8.21. Tampilan Halaman Data Guru Berhasil Diubah 2.....	86
Gambar L8.22. Tampilan Halaman Data Guru Berhasil Dihapus	86
Gambar L8.23. Tampilan Halaman Awal Data Ruang.....	87
Gambar L8.24. Tampilan Halaman Form Data Ruang.....	87
Gambar L8.25. Tampilan Halaman Form Data Ruang Harus Diisi.....	88
Gambar L8.26. Tampilan Halaman Data Ruang Berhasil Disimpan.....	88
Gambar L8.27. Tampilan Halaman Data Ruang Berhasil Diubah.....	89
Gambar L8.28. Tampilan Halaman Data Ruang Berhasil Dihapus.....	89
Gambar L8.29. Tampilan Halaman Awal Data Waktu.....	90
Gambar L8.30. Tampilan Halaman Form Data Waktu.....	90
Gambar L8.31. Tampilan Halaman Form Data Waktu Harus Diisi	91
Gambar L8.32. Tampilan Halaman Data Waktu Berhasil Disimpan 1.....	91
Gambar L8.33. Tampilan Halaman Data Waktu Berhasil Disimpan 2.....	91
Gambar L8.34. Tampilan Halaman Data Waktu Berhasil Diubah 1	92
Gambar L8.35. Tampilan Halaman Data Waktu Berhasil Diubah 2	92
Gambar L8.36. Tampilan Halaman Data Waktu Berhasil Dihapus.....	92
Gambar L8.37. Tampilan Halaman Awal Data Request Waktu.....	93
Gambar L8.38. Tampilan Halaman Form Data Request Waktu.....	93
Gambar L8.39. Tampilan Halaman Form Data Request Waktu Harus Diisi.....	94
Gambar L8.40. Tampilan Halaman Data Request Waktu Berhasil Disimpan.....	94
Gambar L8.41. Tampilan Halaman Data Request Waktu Berhasil Diubah	95

Gambar L8.42. Tampilan Halaman Data Request Waktu Berhasil Dihapus	95
Gambar L8.43. Tampilan Halaman Awal Data Mapel	96
Gambar L8.44. Tampilan Halaman Form Data Mapel	96
Gambar L8.45. Tampilan Halaman Form Data Mapel Harus Diisi	97
Gambar L8.46. Tampilan Halaman Data Mapel Berhasil Disimpan	97
Gambar L8.47. Tampilan Halaman Data Mapel Berhasil Diubah	98
Gambar L8.48. Tampilan Halaman Data Mapel Berhasil Dihapus	98
Gambar L8.49. Tampilan Halaman Awal Data User	99
Gambar L8.50. Tampilan Halaman Form Data User	99
Gambar L8.51. Tampilan Halaman Form Data User Harus Diisi	100
Gambar L8.52. Tampilan Halaman Data User Berhasil Disimpan	100
Gambar L8.53. Tampilan Halaman Data User Berhasil Diubah	101
Gambar L8.54. Tampilan Halaman Data User Berhasil Dihapus	101
Gambar L8.55. Tampilan Halaman Awal Lihat Jadwal	102
Gambar L8.56. Tampilan Halaman Cetak Jadwal	102
Gambar L8.57. Tampilan Halaman Reset Jadwal	103
Gambar L8.58. Laporan Jadwal	103
Gambar L10.1 Hasil Cek Plagiasi Turn It In	110
Gambar L11.1. Kartu Bimbingan	111

DAFTAR TABEL

	Halaman
Tabel 2.1. Penelitian Terdahulu 1	4
Tabel 2.2. Simbol – Simbol Flowchart	7
Tabel 2.3. Simbol Data Flow Diagram	8
Tabel 3.1. Identifikasi Masalah	14
Tabel 3.2. Karakteristik Pengguna	15
Tabel 3.3. Identifikasi Data	15
Tabel 3.4. Analisis Kebutuhan Pengguna	16
Tabel 3.5. Analisis Kebutuhan Fungsional Data Mata Pelajaran.....	16
Tabel 3.6. Analisis Kebutuhan Non-Fungsional	17
Tabel 4.1 Kebutuhan Perangkat Lunak (Software).....	25
Tabel 4.2 Kebutuhan Perangkat Keras (Hardware)	25
Tabel L1.1. Kebutuhan Fungsional Data Ruang.....	51
Tabel L1.2. Analisis Kebutuhan Fungsional Request Waktu	51
Tabel L1.3. Analisis Kebutuhan Fungsional Mapel.....	52
Tabel L1.4. Analisis Kebutuhan Fungsional User	52
Tabel L1.5. Analisis Kebutuhan Fungsional Lihat Jadwal	53
Tabel L1.6. Analisis Kebutuhan Fungsional Jurusan.....	53
Tabel L1.7. Analisis Kebutuhan Fungsional Guru.....	54
Tabel L2.1. Jadwal Kerja	55
Tabel L3.1. Hasil Wawancara	56
Tabel L5.1. Struktur Tabel Guru.....	66
Tabel L5.2. Struktur Tabel Ruang.....	66
Tabel L5.3. Struktur Tabel Waktu	66
Tabel L5.4. Struktur Tabel Program Studi.....	67
Tabel L5.5. Struktur Tabel User	67
Tabel L7.1. Desain Testing Data Waktu.....	74
Tabel L7.2. Desain Testing Data Request Waktu	74
Tabel L7.3. Desain Testing Data Mapel	75
Tabel L7.4. Desain Testing Data User	75
Tabel L7.5. Desain Testing Lihat Jadwal.....	75

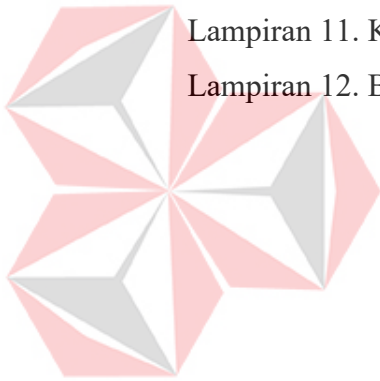
Tabel L7.6. Desain Testing Data Mata Pelajaran	76
Tabel L7.7. Desain Testing Data Jurusan	76
Tabel L7.8. Desain Testing Data Guru	77
Tabel L7.9. Desain Testing Data Ruang	77
Tabel L9.1. Hasil Uji Coba Data Mata Pelajaran.....	104
Tabel L9.2. Hasil Uji Coba Data Jurusan	104
Tabel L9.3. Hasil Uji Coba Data Guru	105
Tabel L9.4. Hasil Uji Coba Data Ruang	106
Tabel L9.5. Hasil Uji Coba Data Waktu	106
Tabel L9.6. Hasil Uji Coba Data Request Waktu	107
Tabel L9.7. Hasil Uji Coba Data Mapel	108
Tabel L9.8. Hasil Uji Coba Data User	108
Tabel L9.9. Hasil Uji Coba Lihat Jadwal.....	109



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Analisis Kebutuhan Fungsional.....	51
Lampiran 2. Jadwal Kerja	55
Lampiran 3. Hasil Wawancara	56
Lampiran 4. System Flow Diagram	57
Lampiran 5. Desain Struktur Database	66
Lampiran 6. Desain I/O.....	68
Lampiran 7. Desain Testing	74
Lampiran 8. Implementasi Sistem.....	78
Lampiran 9. Hasil Uji Coba Sistem	104
Lampiran 10. Hasil Cek Plagiasi.....	110
Lampiran 11. Kartu Bimbingan	111
Lampiran 12. Biodata Penulis	112



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1. Latar Belakang

SMK Negeri 1 Trowulan adalah salah satu sekolah menengah kejuruan yang terletak di daerah kabupaten Mojokerto. SMK Negeri 1 Trowulan memiliki tujuh kompetensi keahlian antara lain agribisnis tanaman pangan dan horticultural, usaha perjalanan wisata, kriya kreatif logam dan perhiasan, teknik komputer dan jaringan, desain komunikasi visual, multimedia, dan teknik pengelasan. Saat ini sekolah memiliki 30 ruangan yang terdiri dari kelas 10 terdapat 10 ruangan, kelas 11 terdapat 10 ruangan, dan kelas 12 terdapat 10 ruangan serta memiliki 5 lab. SMK Negeri 1 Trowulan juga memiliki jam belajar efektif mulai dari jam 08.00 hingga jam 16.00 dan dalam satu tahunnya diadakan 11 kali ujian yang terdiri dari PTS Ganjil, UAS Ganjil, PTS Genap, UAS Genap, PAT (Penilaian Akhir Tahun), Ujian Sekolah, UKK, Uji Kompetensi Keahlian.

Proses pembuatan penjadwalan dimulai Bagian kurikulum membuat penjadwalan, selama pembuatan penjadwalan Bagian Kurikulum kesulitan dalam proses pengecekan ketersediaan ruangan yang digunakan untuk ujian yang dijadwalkan. Selain itu Bagian Kurikulum juga kesulitan dalam menjadwalkan pengawas ujian serta kelas yang mengikuti ujian karena pengawas ujian memiliki beban jaga ujian yang berbeda antara Guru PNS dan Non PNS dan kelas yang mengikuti ujian karena keterbatasan setiap mata pelajaran dibagi menjadi dua sesi dan satu sesinya diikuti oleh 5 kelas saja. Oleh karena itu Bagian Kurikulum harus menyusun ulang kembali dengan cara pengecekan kapasitas ruangan dan jumlah kelas yang mengikuti ujian untuk dipindahkan agar jumlah kapasitas ruangan

terpenuhi dan menyesuaikan beban jaga pengawas dengan jadwal ujian yang sudah dijadwalkan dengan menggunakan Microsoft Excel. Dampak dari permasalahan tersebut membuat penyusunan jadwal ujian membutuhkan waktu yang lama sekitar 1 sampai 2 minggu.

Berdasarkan permasalahan yang dialami SMK 1 Trowulan, maka solusi yang dibutuhkan adalah sistem penjadwalan ujian dengan metode Algoritma Genetika. Selain itu guru juga dapat melihat jadwal menjaga ujian karena mendapatkan akses untuk melihat jadwal ujian. Dengan adanya sistem penjadwalan ujian ini, diharapkan dapat menghasilkan laporan jadwal ujian yang akan diuji dan laporan penjaga ujian.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan, maka rumusan masalahnya adalah bagaimana merancang dan membangun sistem penjadwalan ujian berbasis web pada SMK 1 Trowulan dengan metode Algoritma Genetika.

1.3. Batasan Masalah

Batasan masalah yang dihasilkan dalam pembuatan sistem penjadwalan ujian pada SMK 1 Trowulan adalah sebagai berikut:

1. Sistem ini hanya menjadwalkan ujian dan tidak membahas terkait penjadwalan pembelajaran.
2. Aturan penjadwalan ujian dalam sistem menggunakan kebijakan akademik yang sesuai dengan SMK 1 Trowulan.
3. Jadwal Ujian yang terbentuk pada sistem digunakan oleh kelas 10 sampai dengan kelas 12.
4. Sistem ini hanya menjadwalkan ujian UTS dan UAS.

5. Data yang digunakan adalah data ujian pada sekolah SMK 1 Trowulan mulai periode 2022 sampai 2023.

1.4. Tujuan

Tujuan yang diperoleh berdasarkan rumusan masalah yang telah dijelaskan adalah menghasilkan sistem penjadwalan ujian berbasis web pada SMK 1 Trowulan dengan metode Algoritma Genetika untuk menangani jadwal yang *crash*, akurasi penjadwalan, dan mempercepat waktu pembuatan jadwal ujian.

1.5. Manfaat

Manfaat yang diperoleh dalam pembuatan sistem penjadwalan ujian UTS dan UAS ini adalah sebagai berikut:

1. Wakil Kepala Sekolah (Bagian Kurikulum)
 - a. Mempermudah dalam mengelola data guru data ujian yang akan diuji.
 - b. Mempermudah dalam membuat jadwal ujian tanpa banyak proses yang dilakukan.
 - c. Mempermudah dalam pengaturan jam menjaga ujian antara Guru PNS dan Non PNS.
 - d. Meminimalkan kesalahan dalam hasil penjadwalan ujian.
 - e. Mengurangi waktu dalam pembuatan jadwal ujian.
2. Kepala Sekolah untuk mempermudah dalam memberikan persetujuan untuk jadwal ujian yang telah dibuat.
3. Guru dapat mengetahui jadwal menjaga ujian.

BAB II

LANDASAN TEORI

2.1. Penelitian Terdahulu

Penelitian terdahulu merupakan salah satu pedoman penulis dalam melakukan penelitian ini agar penulis dapat mengetahui teori yang digunakan yang dapat dilihat pada tabel 2.1.

Tabel 2.1. Penelitian Terdahulu 1

Nama Peneliti	Judul Penelitian	Hasil Penelitian
(Mahendra, Cobantoro, & Nurwanto, 2022)	Rancang Bangun Aplikasi <i>management</i> Penjadwalan Perkuliahan Dengan Algoritma Genetika Berbasis Website	Aplikasi <i>management</i> penjadwalan yang telah dibuat dapat membantu Universitas dalam membuat jadwal tanpa adanya jadwal yang bentrok karena penjadwalan dioptimalkan dengan metode algoritma genetika.
Perbedaan: Pada penelitian ini studi kasusnya penjadwalan perkuliahan pada universitas yang subjeknya dilihat dari bobot sks tiap mahasiswa, beban dosen, dan bobot per kelas. Sedangkan pada penelitian ini penjadwalan ujian pada SMK yang subjectnya dilihat dari jurusan, beban guru, dan kelas. Pada Penelitian ini menggunakan indikator mata kuliah dan dosen dalam penentuan penjadwalan perkuliahan, sedangkan dalam penelitian yang dilakukan penulis menggunakan indikator mata pelajaran, guru, kelas, dan kejuruan dalam penentuan penjadwalan ujian.		
(Sasongko, 2020)	Rancang Bangun Aplikasi Penjadwalan Mata Pelajaran Pada SMA Giki 2 Surabaya	Aplikasi penjadwalan mata pelajaran ini berhasil membentuk jadwal mata pelajaran yang sesuai dan tanpa adanya yang bentrok dan laporan yang dihasilkan adalah laporan jadwal mata pelajaran dan laporan rincian beban mengajar.
Perbedaan: Pada penelitian ini melakukan penjadwalan mata pelajaran pada SMA yang dimana SMA hanya memiliki dua jurusan yaitu IPA dan IPS, sedangkan dalam penelitian yang dilakukan penulis adalah penjadwalan ujian pada SMK dengan Sembilan jurusan. Pada penelitian ini menghasilkan laporan jadwal mata pelajaran dan laporan rincian beban mengajar, sedangkan dalam penelitian yang dilakukan penulis adalah menghasilkan laporan jadwal ujian dan laporan rincian penjaga ujian.		
(Putranto, Utami, & Sunyoto, 2017)	Perancangan Sistem Penjadwalan Ujian Menggunakan Algoritma Genetika Pada STMIK AMIKOM Purwokerto	Aplikasi penjadwalan ujian menggunakan algoritma genetika mengalokasikan ruangan dan pengawas ujian lebih merata dengan indikator nilai standar deviasi data stimulasi lebih kecil dari data real pada hasil <i>generate</i> data UAS Gasal dan UTS Genap 2015/2016.
Perbedaan : Pada penelitian ini menggunakan algoritma genetika untuk alokasi ruangan dan pengawas agar lebih merata, sedangkan untuk penelitian ini pengalokasiannya lebih fokus agar ruangan, pengawas, kelas per jurusan lebih merata.		

2.2. Penjadwalan

Menurut Pinedo (2012), penjadwalan dapat didefinisikan sebagai proses pengalokasian sumber daya untuk mengerjakan sekumpulan tugas dalam jangka waktu tertentu dengan 2 arti penting sebagai berikut:

1. Penjadwalan merupakan suatu fungsi pengambilan keputusan untuk membuat atau menentukan jadwal.
2. Penjadwalan merupakan suatu teori yang berisi sekumpulan prinsip dasar, model, teknik, dan kesimpulan logis dalam proses pengambilan keputusan yang memberikan pengertian dalam fungsi.

Sedangkan Menurut Farida (Farida, 2008) penjadwalan merupakan suatu kumpulan kebijakan dan mekanisme dalam sistem operasi yang berhubungan dengan urutan kerja yang dilakukan sistem komputer. Penjadwalan digunakan untuk memutuskan proses yang harus bekerja serta kapan dan berapa lama proses tersebut berjalan. Sasaran utama proses penjadwalan adalah:

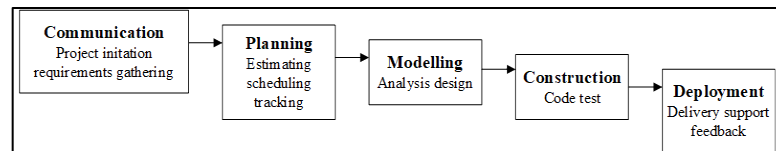
Penjadwalan ujian pada SMK 1 Trowulan memiliki beberapa aturan - aturan yang harus diterapkan, antara lain:

1. Jadwal PTS untuk mata pelajaran yang sama dilaksanakan ada hari yang sama.
2. Mata pelajaran jurusan disesuaikan dengan jurusan masing – masing.
3. Untuk satu sesi hanya ada 1 pengawasan tiap 1 ruangan.

2.3. *System Development Life Cycle*

Menurut Pressman (2015) *System development life cycle* (SDLC) ini biasanya disebut dengan model *waterfall*. Menurut Pressman (2015), nama lain dari Model *waterfall* adalah model air terjun terkadang dinamakan siklus hidup klasik

(*classic life cycle*), dimana hal ini menyiratkan pendekatan yang sistematis dan berurutan (sekuensial) pada pengembangan perangkat lunak. Pengembangan perangkat lunak dimulai dari spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan – tahapan perencanaan (*planning*), pemodelan (*modelling*), konstruksi (*construction*), dan *deployment*.



Gambar 2.1. Tahapan *System Development Life Cycle* model *waterfall*

Berikut ini adalah penjelasan dari tahap – tahap yang dilakukan di dalam model *waterfall* (Pressman, 2015):

A. *Communication*

Langkah ini merupakan analisis terhadap kebutuhan *software* dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan *customer*, maupun mengumpulkan data - data tambahan baik yang ada di jurnal, artikel maupun dari internet.

B. *Planning*

Proses ini merupakan lanjutan dari proses *communication* (*analysis requirement*).

C. *Modelling*

Proses ini menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada rancangan struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*.

D. Construction

Proses pembuatan kode. *Coding* atau pengkodean merupakan penerjemah desain dalam bahasa yang bisa dikenali oleh komputer. *Programmer* akan menerjemahkan transaksi.

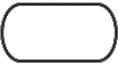

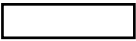



E. Deployment


Tahapan ini bisa dikatakan akhir dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

2.4. Bagan Alir

Bagan Alir / *flowchart* yang digunakan oleh penulis dalam penelitian ini yaitu untuk menunjukkan prosedur untuk membuat sebuah jadwal pada SMK 1 Trowulan. Menurut Yakub (2012), *Flowchart* atau Bagan alir adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir (*flowchart*) digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

Tabel 2.2. Simbol – Simbol Flowchart



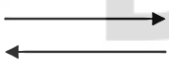

No	Simbol	Nama Simbol	Fungsi
1		<i>Terminator</i>	Permulaan atau akhir program
2		Garis Alir	Arah aliran program
3		Proses	Proses perhitungan data
4		<i>Input/Output Data</i>	Proses <i>input</i> atau <i>output</i> data
5		<i>Decision</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
6		<i>On Page Connector</i>	Penghubung bagian - bagian <i>flowchart</i> yang berada pada halaman yang sama

No	Simbol	Nama Simbol	Fungsi
7		<i>Off Page Connector</i>	Penghubung bagian - bagian <i>flowchart</i> yang berada pada halaman berbeda

2.5. Data Flow Diagram

Menurut Sutabri (2012), *Data Flow Diagram* adalah suatu *network* yang menggambarkan suatu *system automatic* atau komputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai aturan mainnya. Beberapa poin penting di dalam DFD adanya suatu entitas, proses, arus data, dan data store (Ridwan, Kustian, & Ambarsa, 2022).

Tabel 2.3. Simbol Data Flow Diagram

No	Simbol	Keterangan
1		Proses, Simbol ini digunakan untuk proses pengolahan atau transformasi data.
2		<i>External Entity</i> , Simbol ini digunakan untuk menggambarkan asal atau tujuan data.
3		<i>Data Flow</i> , simbol ini digunakan untuk menggambarkan aliran data yang berjalan.
4		<i>Data Store</i> , Simbol ini digunakan untuk menggambarkan data <i>flow</i> yang sudah disimpan atau diarsipkan.

2.6. Algoritma Genetika

Algoritma genetika merupakan evaluasi atau perkembangan dunia komputer dalam bidang kecerdasan buatan (*artificial intelligence*). Kemunculan algoritma genetika ini terinspirasi oleh teori Darwin dan teori-teori dalam ilmu biologi, sehingga banyak istilah dan konsep biologi yang digunakan dalam algoritma genetika, karena sesuai dengan namanya, proses-proses yang terjadi dalam algoritma genetika sama dengan apa yang terjadi pada evaluasi biologi (Suyatno, 2014). Adapun Komponen-Komponen dalam Algoritma Genetika yaitu:

1. Skema Pengkodean

Teknik pengkodean adalah bagaimana mengkodekan gen dari kromosom, gen merupakan bagian dari kromosom. Satu gen akan mewakili satu variable (guru, kelas, sesi ujian, hari). Gen dapat direpresentasikan dalam bentuk: bit, bilangan real, string, daftar aturan, gabungan dari beberapa kode, elemen permutasi, elemen program atau representasi lainnya yang dapat diimplementasikan untuk operator genetika dalam kasus ini menggunakan elemen program berupa id setiap variable.

2. Membangkitkan Populasi Awal dan Kromosom

Membangkitkan populasi awal adalah proses membangkitkan sejumlah individu atau kromosom secara acak (*random*).

3. Nilai *Fitness*

Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran performansinya. Di Dalam evolusi alam, individu yang bernilai *fitness* tinggi yang akan bertahan hidup. Sedangkan individu yang bernilai *fitness* rendah akan mati.

Fungsi *fitness* tergantung pada permasalahan tertentu dari representasi yang digunakan. Sehingga perhitungan nilai *fitness*-nya menjadi:

$$f=1/(1+F_1B_1+F_2B_2\dots) \dots\dots\dots(1)$$

Keterangan:

B_n = Bobot Pelanggaran.

F_n = Banyaknya Pelanggaran.

4. Seleksi

Pembentukan susunan kromosom pada suatu populasi baru biasanya dilakukan secara proporsional sesuai dengan nilai *fitness*-nya. Suatu metode seleksi

yang umumnya digunakan adalah *roulette wheel*. Metode *roulette wheel* adalah pemilihan kromosom dengan probabilitas nilai *fitness* tertinggi.

5. Pindah Silang (*Crossover*)

Salah satu komponen yang paling penting dalam algoritma genetika adalah pindah silang atau *crossover*. Sebuah kromosom yang mengarah pada solusi yang baik dapat diperoleh dari proses memindah-silangkan dua buah kromosom untuk menghasilkan keturunan kromosom yang baru.

6. Mutasi

Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam suatu kromosom. Mutasi ini berperan untuk menggantikan gen yang hilang dari populasi akibat seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi. Proses mutasi dalam pengkodean nilai dapat dilakukan dengan cara, yaitu dengan memilih acak posisi gen pada kromosom, nilai yang ada tersebut kemudian diubah dengan suatu nilai tertentu yang diambil secara acak. Contoh: Kromosom sebelum mutasi : 1 3 4 7 6, Kromosom sesudah mutasi :

1 2 4 8 6.

7. Elitisme

Elitisme merupakan proses penyimpanan kromosom atau individu untuk menjaga agar individu bernilai *fitness* tertinggi tersebut tidak hilang selama evolusi, maka perlu dibuat beberapa salinannya.

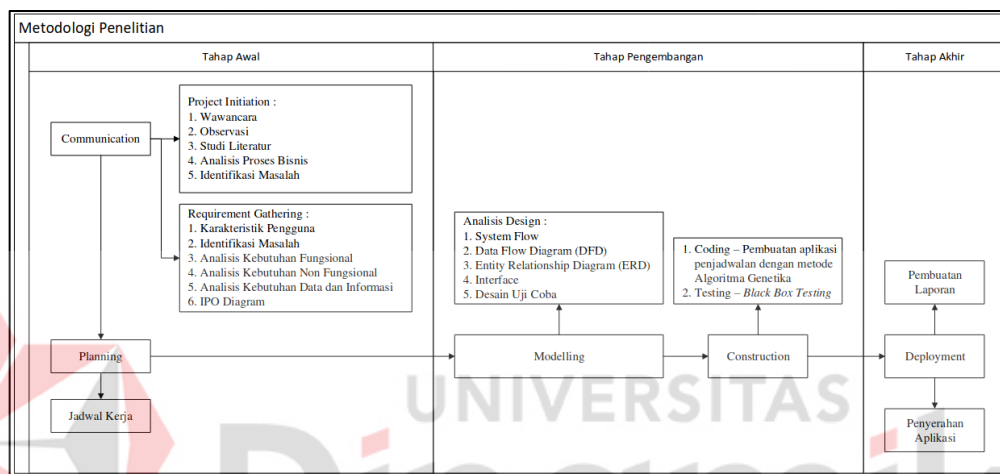
8. Kondisi Selesai

Kondisi selesai yang dapat menghentikan proses algoritma genetika ini adalah jika jumlah generasi atau iterasi maksimum telah tercapai yaitu nilai *fitness* bernilai 1.

BAB III

METODOLOGI PENELITIAN

Metodologi Penelitian yang digunakan untuk pembuatan sistem penjadwalan ini dapat dilihat pada Gambar 3.1, Tahapan – tahapannya terdiri dari tahapan awal, pengembangan, dan akhir.



Gambar 3.1. Metodologi Penelitian

1.1. Communication

Tahapan yang akan dilakukan pertama kali dalam penelitian adalah *Communication*. Dalam tahapan *Communication* ini mempunyai dua sub tahapan yaitu *Project Initiation* dan *Requirement Gathering*.

1.1.1. Project Initiation

Dalam sub tahapan *Project Initiation* ini merupakan tahapan untuk mengumpulkan data – data yang diperlukan. Langkah – Langkah yang digunakan untuk mengumpulkan data adalah sebagai berikut:

A. Wawancara

Wawancara telah dilakukan pada tanggal 01 September 2022 dengan pihak sekolah SMK 1 Trowulan yaitu Bapak Agus selaku bagian akademik atau wakil

kepala sekolah. Tujuan dari wawancara ini adalah untuk mengetahui alur proses penjadwalan ujian saat ini dan permasalahan yang dihadapi selama membuat jadwal ujian.

B. Observasi

Observasi telah dilakukan pada SMK 1 Trowulan dengan melihat dan mengamati secara langsung dalam proses pembuatan jadwal ujian yang dilakukan oleh wakil kepala sekolah. Observasi ini menghasilkan data – data pendukung yang dibutuhkan untuk pembuatan aplikasi.

C. Studi Literatur

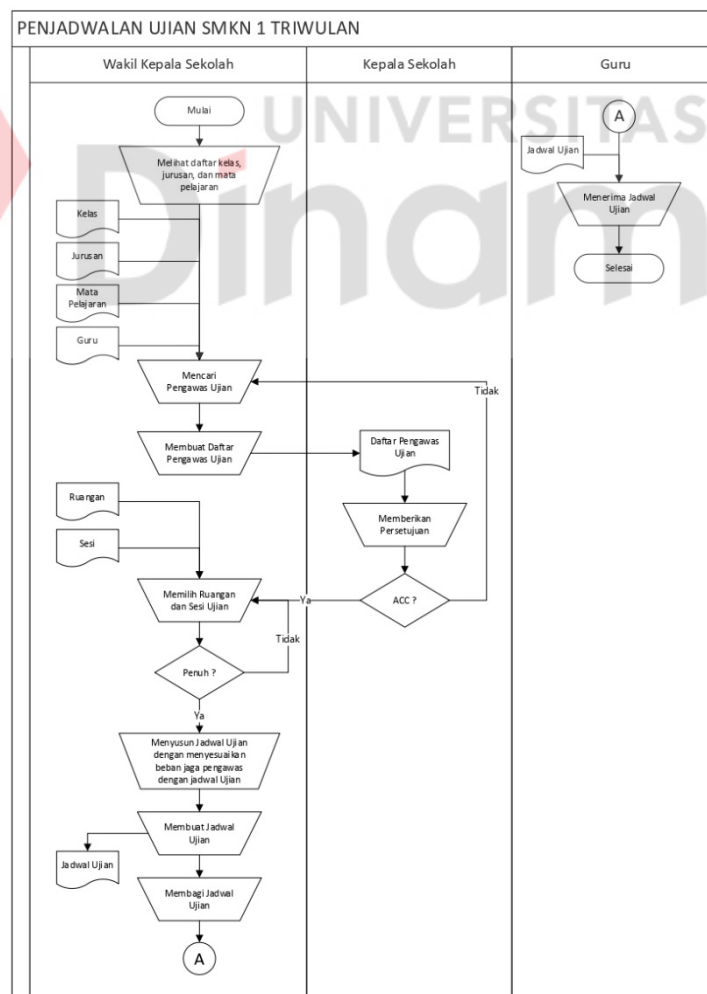
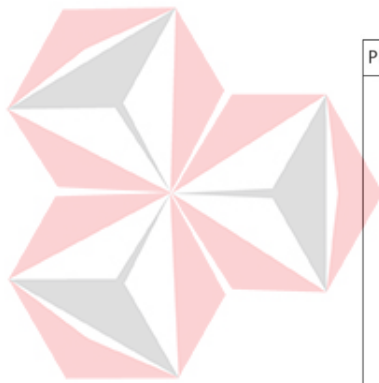
Studi Literatur diperlukan agar penelitian ini mendapatkan kajian teori agar memperdalam wawasan untuk mendapatkan solusi yang optimal dari beberapa jurnal ilmiah dan buku para ahli yang melakukan penelitian pada bidang yang sama.

1. *SDLC*
2. Penjadwalan
3. *Blackbox Testing*
4. Algoritma Genetika
5. Bagan Alir
6. *Data Flow Diagram*

D. Analisis Proses Bisnis

Proses bisnis digunakan untuk mengetahui alur yang terjadi sekarang untuk menganalisis pembuatan aplikasi penjadwalan ujian. Proses penjadwalan ujian pada saat ini yang terjadi pada SMK 1 Trowulan adalah Pada Setiap ujian Wakil

Kepala Sekolah melihat daftar kelas, jurusan, dan mata pelajaran. Kemudian Wakil Kepala Sekolah mencari pengawas ujian dan dilanjutkan untuk membuat daftar program ujian. Kemudian wakil kepala sekolah memberikan daftar program ujian kepada kepala sekolah. Dan kepala sekolah memberikan persetujuan kepada wakil kepala sekolah jika tidak wakil kepala sekolah harus membuat program ujian ulang. Kemudian setelah disetujui wakil kepala sekolah memilih ruangan dan sesi ujian. Setelah selesai menentukan ruangan dan sesi ujian wakil kepala sekolah mulai menyusun beban jaga pengawas dengan jadwal ujian. Setelah jadwal ujian selesai wakil kepala sekolah membagikan kepada guru - guru pengawas ujian yang dapat dilihat pada gambar 3.2.



Gambar 3.2. Proses Bisnis

E. Identifikasi Masalah

Pada tahap ini dilakukan identifikasi masalah berdasarkan analisis dan wawancara yang telah dilakukan sebelumnya, berikut identifikasi masalah dan solusi yang dihasilkan pada tabel 3.1. di bawah ini

Tabel 3.1. Identifikasi Masalah

No	Permasalahan	Dampak	Solusi
1.	Bagian Kurikulum kesulitan dalam proses pengecekan ketersediaan ruangan yang digunakan untuk ujian yang dijadwalkan.	Jadwal yang telah dibuat tidak sesuai dengan indikator yang dibutuhkan.	sistem pengelolaan ruangan yang cepat dalam memproses data menjadi sebuah informasi.
2.	Bagian Kurikulum juga kesulitan dalam menjadwalkan pengawas ujian serta kelas yang mengikuti ujian karena pengawas ujian memiliki beban jaga ujian yang berbeda antara Guru PNS dan Non PNS dan kelas yang mengikuti ujian karena keterbatasan setiap mata pelajaran dibagi menjadi dua sesi dan satu sesinya diikuti oleh 5 kelas saja.	Bagian Kurikulum harus menyusun ulang kembali dengan cara pengecekan kapasitas ruangan dan jumlah kelas yang mengikuti ujian untuk dipindahkan agar jumlah kapasitas ruangan terpenuhi dan menyesuaikan beban jaga pengawas.	Sistem Pengolahan data ketersediaan guru.
3.	jika jadwal mengalami sebuah masalah penyusunan ulang jadwal ujian membutuhkan waktu yang lama sekitar 2 minggu.	Bagian Kurikulum harus menyusun ulang kembali dengan cara pengecekan kapasitas ruangan dan jumlah kelas yang mengikuti ujian untuk dipindahkan agar jumlah kapasitas ruangan terpenuhi dan menyesuaikan beban jaga pengawas.	Sistem Penjadwalan yang dapat meningkatkan akurasi pembuatan penjadwalan ujian.

1.1.2. Requirement Gathering

Requirement Gathering adalah suatu aktivitas yang dilakukan untuk menganalisa kebutuhan fungsional dari aplikasi ke tahap desain.

A. Karakteristik Pengguna

Identifikasi pengguna digunakan untuk mengetahui siapa saja pengguna aplikasi yang akan dikembangkan yaitu Sistem Penjadwalan pada SMK 1

Trowulan. Pada SMK 1 Trowulan dalam melakukan proses penjadwalan teridentifikasi beberapa pengguna yang terlibat diantaranya yaitu:

Tabel 3.2. Karakteristik Pengguna

No.	Pengguna	Fungsi	Informasi
1.	Admin	1. Mengelola data guru 2. Mengelola data ruang 3. Mengelola data mata pelajaran 4. Mengelola data user 5. Mengelola data waktu 6. Mengelola data request waktu 7. Mengelola data jurusan 8. <i>Generate</i> jadwal	1. Informasi data guru 2. Informasi data ruang 3. Informasi data mata pelajaran 4. Informasi data user 5. Informasi data waktu 6. Informasi data request waktu 7. Informasi data jurusan 8. Informasi jadwal
2.	Guru, Kepala Sekolah, Siswa	Melihat Jadwal	Informasi data jadwal

B. Identifikasi Data

Identifikasi kebutuhan data sangat diperlukan karena data-data yang ada merupakan gambaran pada sistem yang akan dibuat nantinya, berikut dapat dilihat pada tabel 3.3 di bawah ini:

Tabel 3.3. Identifikasi Data

Jenis Data	Nama Data
<i>Master</i>	1. Data Mata Pelajaran 2. Data User 3. Data Ruang 4. Data Waktu
<i>Transaksi</i>	1. Data Jadwal Pelajaran

A. Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna adalah analisis yang bertujuan untuk mengetahui kebutuhan fungsi, data, dan informasi apa saja nanti yang ada dalam aplikasi. Berikut merupakan hasil analisis kebutuhan pengguna.

Tabel 3.4. Analisis Kebutuhan Pengguna

No	Pengguna	Fungsional	Data	Informasi
1	Admin	Pengelolaan Data Master	Mata Pelajaran	Daftar informasi mata pelajaran
			Jurusan	Daftar informasi jurusan
			Guru	Daftar informasi guru
			Ruang	Daftar informasi ruang
			Waktu	Daftar informasi waktu
			Request waktu	Daftar informasi request waktu
			Mapel	Daftar informasi mapel
			User	Daftar informasi user
		Buat Jadwal	Waktu, guru, mata pelajaran, mapel, jurusan	Daftar Jadwal pelajaran
		Lihat Jadwal	Data Mapel	Daftar Jadwal pelajaran
2	Guru, Admin, Siswa, Kepala Sekolah	Lihat Jadwal	Data Mapel	Daftar Jadwal pelajaran

B. Analisis Kebutuhan Fungsional

Berikut ini merupakan analisis kebutuhan fungsional dari setiap fungsi yang ada dalam aplikasi penjadwalan SMK 1 Trowulan dapat dilihat pada Lampiran 1.

Tabel 3.5. Analisis Kebutuhan Fungsional Data Mata Pelajaran

Nama Fungsi	Fungsi Data Master Mata Pelajaran	
Deskripsi	Fungsi ini merupakan proses untuk pengelolaan data <i>master</i> mata pelajaran	
Kondisi Awal	Tabel <i>Master</i> mata pelajaran belum terisi	
Alur Normal	Aksi Pengguna	Alur Normal
	<i>Admin</i> memilih menu <i>master</i> mata pelajaran	Sistem menampilkan data <i>master</i> mata pelajaran
	Menambah Data Mata Pelajaran	
	<i>Admin</i> menekan tombol <i>input</i> data mata pelajaran	Sistem menampilkan <i>form</i> data mata pelajaran
	<i>Admin</i> mengisi <i>form</i> data <i>master</i> mata pelajaran	<i>Field</i> <i>form</i> data mata pelajaran telah terisi
	<i>Admin</i> menekan tombol simpan	Data mata pelajaran telah tersimpan kedalam <i>database</i> dan sistem menampilkan halaman <i>master</i> mata pelajaran
	Mengubah Data Mata Pelajaran	
	<i>Admin</i> menekan tombol <i>edit</i>	Sistem menampilkan <i>form</i> <i>edit</i> data mata pelajaran
	<i>Admin</i> mengisi <i>form</i> <i>edit</i> data <i>master</i> mata pelajaran	<i>Field</i> <i>form</i> <i>edit</i> data mata pelajaran telah terisi
	<i>Admin</i> menekan tombol simpan	Data mata pelajaran telah diperbarui pada <i>database</i> dan sistem menampilkan halaman <i>master</i> mata pelajaran
	Hapus Data Mata Pelajaran	
	<i>Admin</i> menekan tombol <i>delete</i>	Data mata pelajaran telah dihapus pada <i>database</i> dan sistem menampilkan halaman <i>master</i> mata pelajaran

Nama Fungsi	Fungsi Data Master Mata Pelajaran
Kondisi Akhir	Tabel <i>mastermata</i> pelajaran telah terisi

C. Analisis Kebutuhan Non-Fungsional

Berikut ini merupakan analisis kebutuhan non-fungsional aplikasi penjadwalan SMK 1 Trowulan.

Tabel 3.6. Analisis Kebutuhan Non-Fungsional

No	Analisis Kebutuhan Non Fungsional		
1	Sistem operasi yang digunakan minimal windows 7 keatas dengan komputer <i>client</i> 32 bit dan server 64 bit		
		Server	Client
2	Waktu Respon	Respon waktu yang dibutuhkan untuk mengakses aplikasi kemungkinan mempunyai <i>delay</i> 4 detik.	Respon waktu yang dibutuhkan untuk mengakses aplikasi sesuai dengan hak-hak atau otoritas akses masing masing kemungkinan mempunyai <i>delay</i> paling lama 30 detik
3	Storage	Penyimpanan yang dibutuhkan untuk server yang digunakan sekitar 35gb sistem operasi sebesar 20 gb, MySQL sebesar 10 gb, Microsoft Office 2 gb, 150 mb Xampp dan 250 mb Firefox	Penyimpanan yang dibutuhkan untuk <i>client</i> yang digunakan sekitar 35gb sistem operasi sebesar 20 gb, MySQL sebesar 10 gb, Microsoft Office 2 gb, 150 mb Xampp dan 250 mb Firefox

1.2. Planning

Tahapan yang akan dilakukan untuk perencanaan estimasi waktu dalam pembuatan aplikasi merupakan tahapan *planning*.

1.3. Modelling

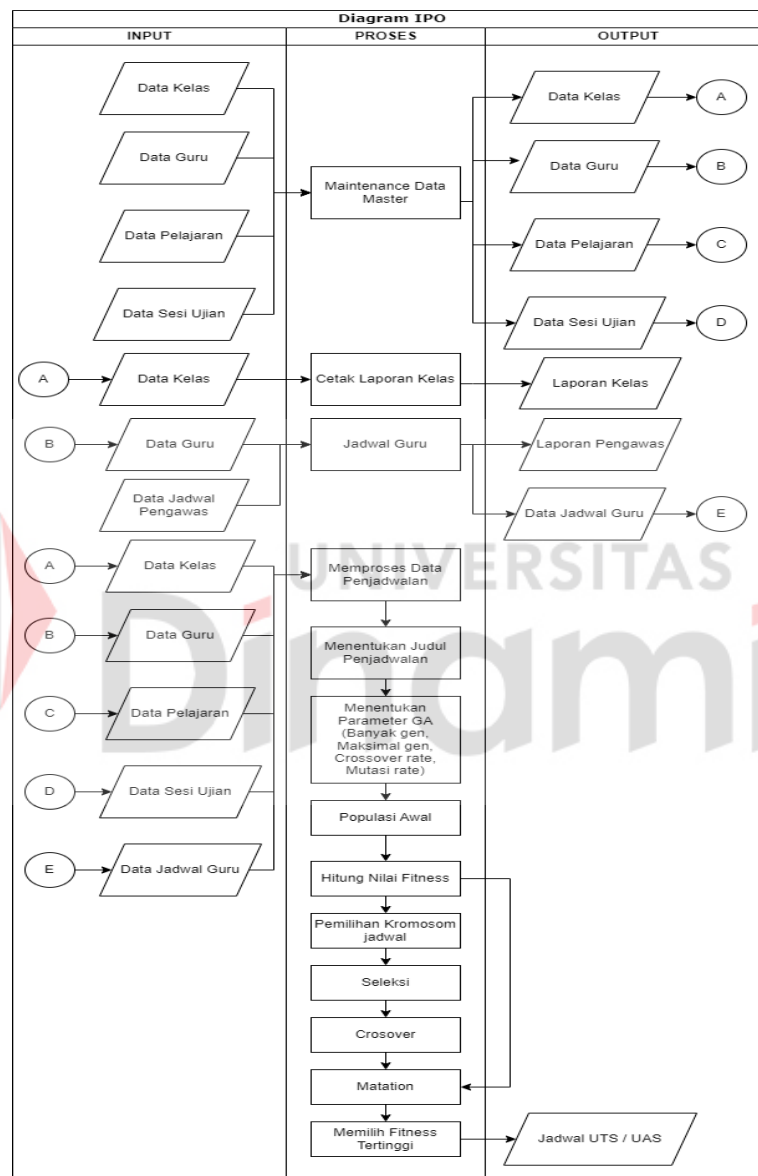
Tahapan ini *modelling* dibagi menjadi dua sub tahapan yaitu analisis sistem dan perancangan sistem. Analisis sistem terdapat beberapa tahapan antara lain *System Flow*, *Data Flow Diagram* (DFD), dan *Entity Relationship Diagram* (ERD).

Pada perancangan sistem juga memiliki beberapa tahapan yaitu *interface* dan uji coba.

1.3.1. Process Model

A. Desain *Input Output*

Berikut ini merupakan Desain *Input* dan *Output* aplikasi penjadwalan pada SMK 1 Trowulan.



Gambar 3.3. IPO Diagram

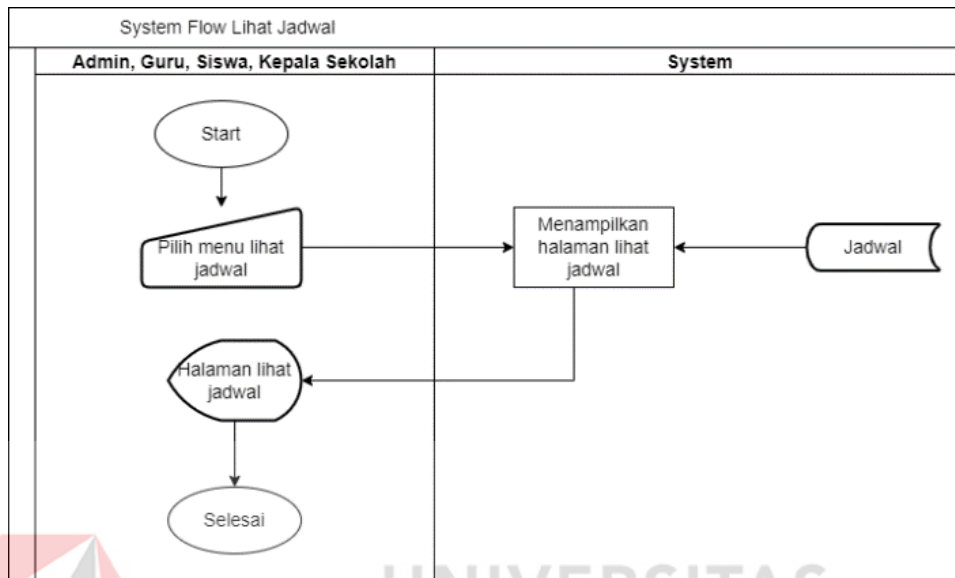
B. System Flow

System flow merupakan sebuah alur proses aplikasi penjadwalan pada SMK

1 Trowulan. *System flow* lainnya dapat dilihat pada lampiran 4.

1.1. Guru, Siswa dan Kepsek

System Flow Lihat Jadwal pada Aplikasi Penjadwalan merupakan fungsi yang menjelaskan alur aplikasi dalam menampilkan jadwal ujian yang telah di *generate* yang dapat dilihat pada Gambar 3.8



Gambar 3.4. System Flow Diagram Lihat Jadwal

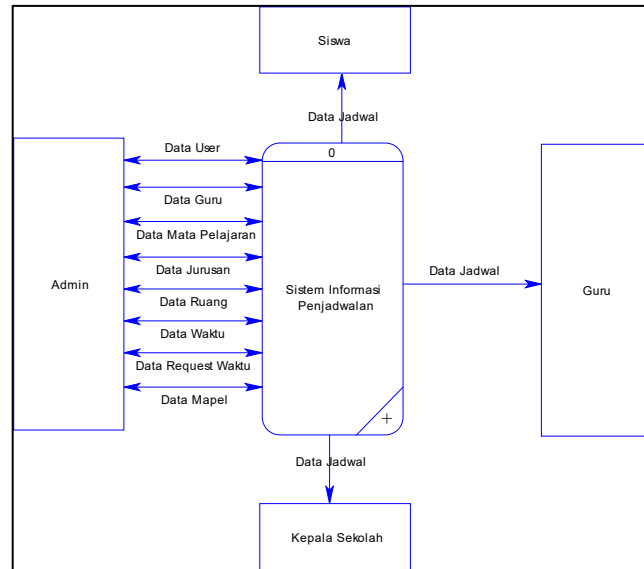
1.3.2. Data Model

A. Data Flow Diagram

Data flow diagram merupakan sebuah diagram dalam menggambarkan aliran data pada tiap entitas.

1. Context Diagram

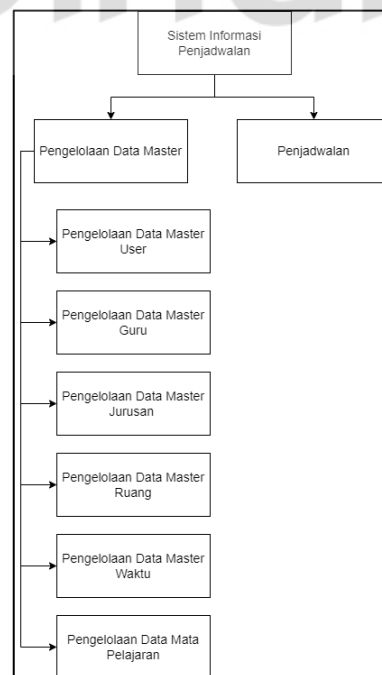
Context diagram merupakan sebuah diagram yang menggambarkan ruang lingkup sistem serta keseluruhan sistem. aplikasi penjadwalan pada SMK 1 Trowulan yang terdapat empat entitas yaitu kepala sekolah, guru, murid atau siswa, dan admin.



Gambar 3.5. Context Diagram

2. Diagram Berjenjang

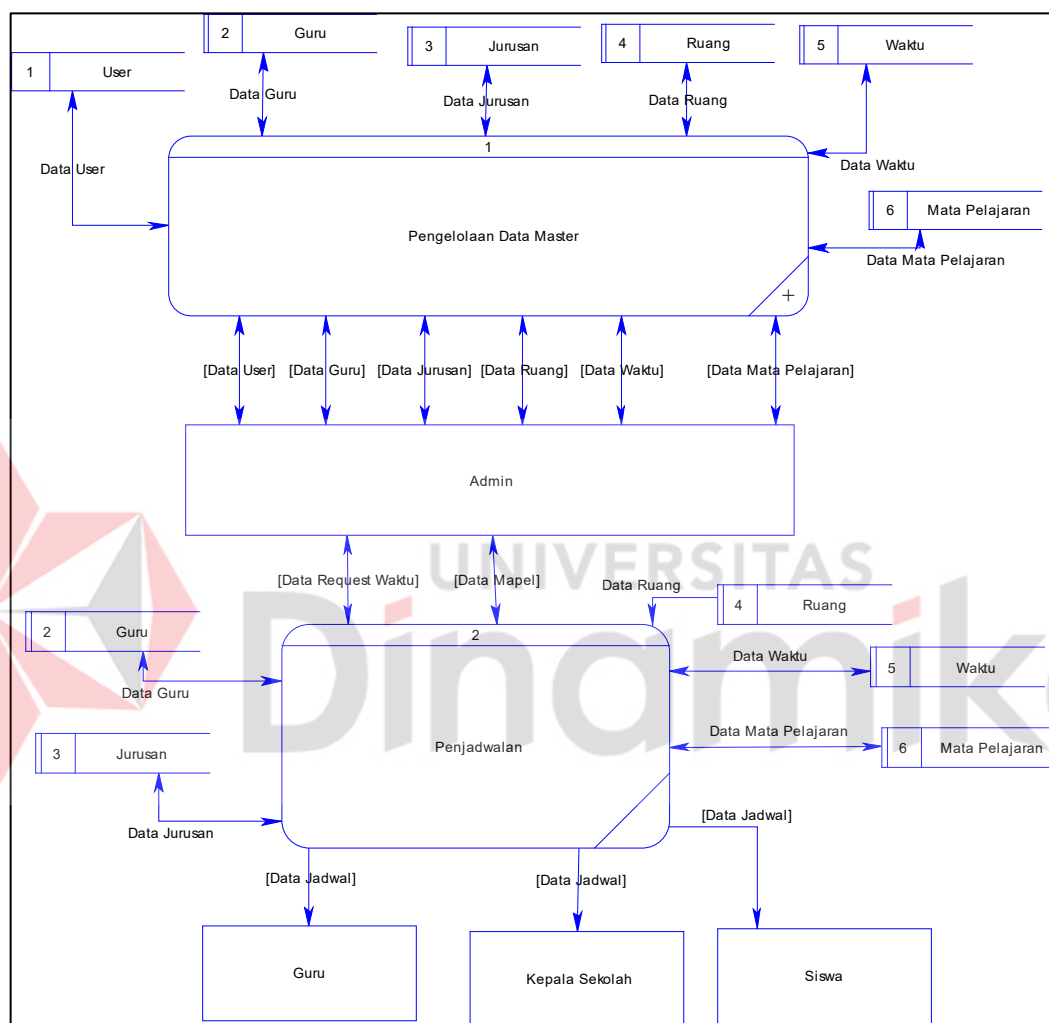
Diagram jenjang merupakan bagan jenjang yang menggambarkan sebuah struktur sistem dalam semua proses yang didapatkan dari *context* diagram. Berikut merupakan diagram jenjang yang dapat dilihat pada Gambar 3.6. Diagram Berjenjang



Gambar 3.6. Diagram Berjenjang

3. Data Flow Diagram Level 0

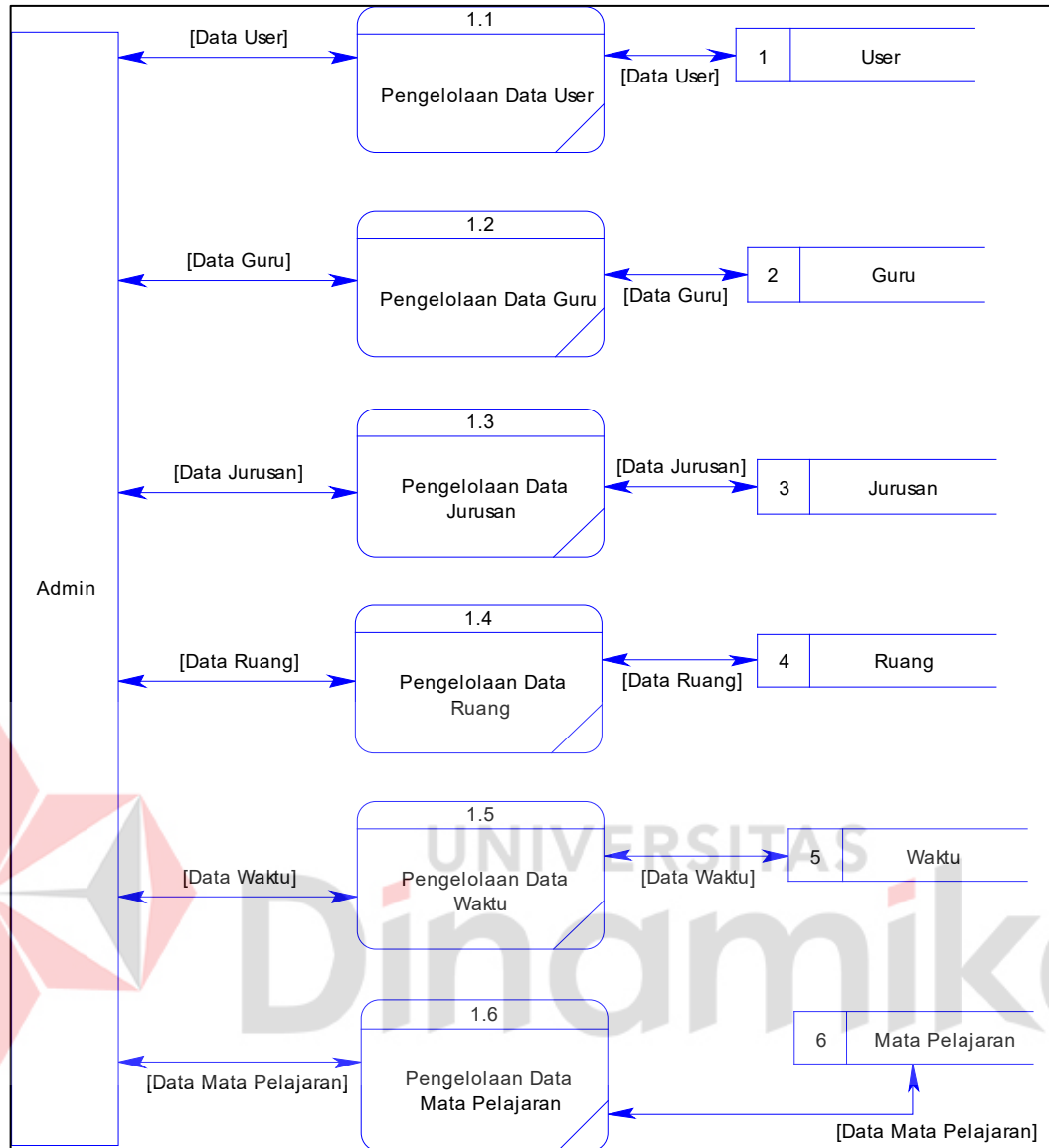
Pada *data flow diagram level 0* merupakan hasil *decompose* dari *context diagram* dengan menjelaskan detail keseluruhan sistem. *Data flow diagram level 0* dapat dilihat pada Gambar 3.7. *Data Flow Diagram Level 0*



Gambar 3.7. *Data Flow Diagram Level 0*

4. *Data Flow Diagram Level 1 Data Master*

Pada *data flow diagram level 1 master data* merupakan hasil *decompose* dari *Data Flow Diagram Level 0* pada *master data*. Berikut merupakan *data flow diagram level 1* yang dapat dilihat pada gambar 3.12.

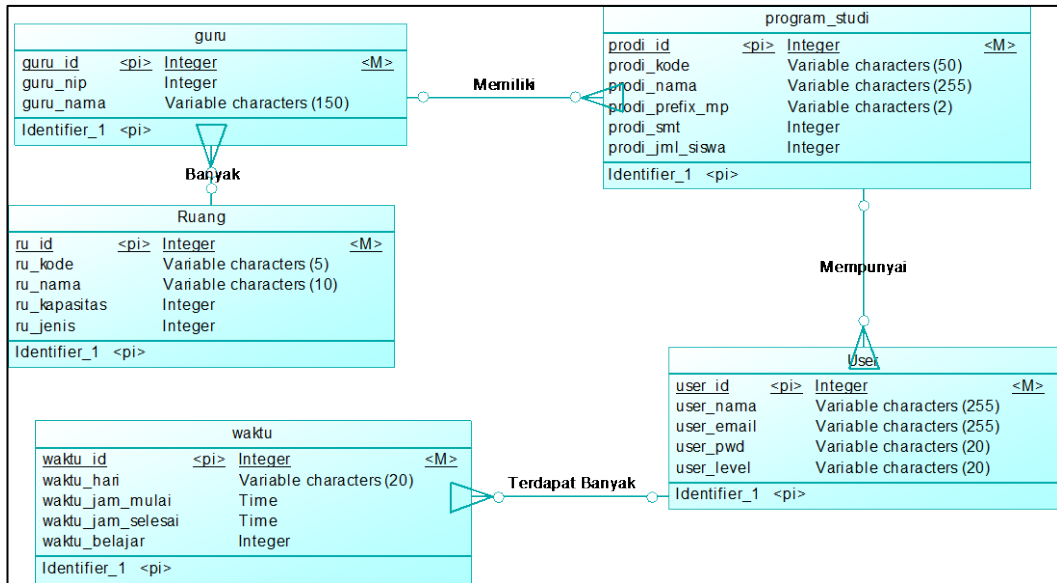


Gambar 3.8. Data Flow Diagram Level 1 Data Master

B. Entity Relationship Diagram

1. Conceptual Data Model

Conceptual Data Model dalam aplikasi Penjadwalan pada SMK 1 Trowulan yang memiliki lima data tabel.

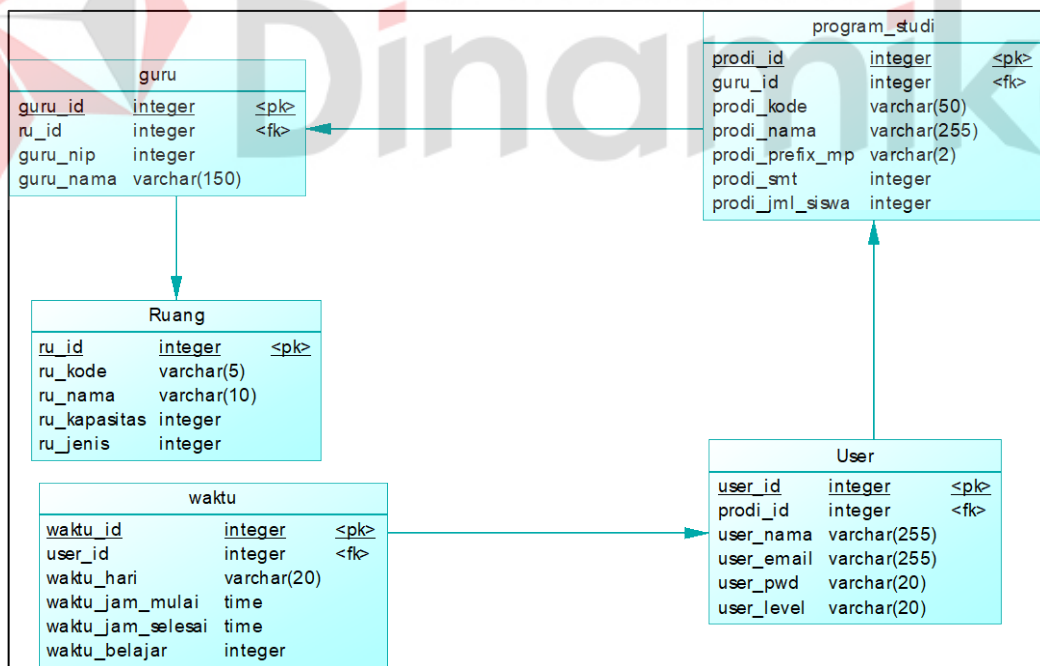


Gambar 3.9. Conceptual Data Model

2. Physical Data Model

Physical Data Model pada aplikasi penjadwalan pada SMK 1 Trowulan.

Untuk lebih jelasnya *physical Data Model* dapat dilihat pada gambar berikut ini.



Gambar 3.10. Physical Data Model

1.3.3. Desain Sistem

A. Desain Struktur Database

Struktur tabel aplikasi penjadwalan SMK 1 Trowulan yang dapat dilihat pada Lampiran 5.

B. Desain *Interface*

Desain *Interface* aplikasi penjadwalan SMK 1 Trowulan yang dapat dilihat pada Lampiran 6.

C. Desain *Testing*

Desain *testing* merupakan perencanaan uji coba sistem yang akan dibuat untuk pengujian fungsi pada sistem. Berikut merupakan desain testing penjadwalan.

Untuk desain *testing* lengkapnya dapat dilihat pada lampiran 7.



UNIVERSITAS
Dinamika

BAB IV
HASIL DAN PEMBAHASAN

4.1 Kebutuhan Sistem

4.1.1 Spesifikasi Sistem

Spesifikasi sistem merupakan informasi yang berhubungan dengan kebutuhan yang diperlukan dalam melakukan pengimplementasian sistem yang sudah dirancang.

A. Kebutuhan Perangkat Lunak (*Software*)

Berikut merupakan rincian spesifikasi perangkat lunak (*software*) yang dibutuhkan dapat dilihat pada Tabel 4.1.

Tabel 4.1 Kebutuhan Perangkat Lunak (*Software*)

Software	Keterangan
XAMPP	<i>Web Server Local</i>
Notepad++	<i>Text Editor</i>
Mysql	<i>Database Server</i>
Chrome	<i>Web Browser</i>
Windows 10 Pro	<i>Sistem Operasi</i>

B. Kebutuhan Perangkat Keras (*Hardware*)

Berikut merupakan rincian spesifikasi perangkat keras (*hardware*) yang dibutuhkan dapat dilihat pada Tabel 4.2.

Tabel 4.2 Kebutuhan Perangkat Keras (*Hardware*)

Hardware	Spesifikasi
<i>Processor</i>	Intel Core i3
RAM	2 gb
<i>Disk Drive</i>	500 gb
Modem	<i>Speed min. 2 Mbps</i>
<i>I/O Devices</i>	<i>Monitor atau LCD, Mouse, dan Keyboard</i>

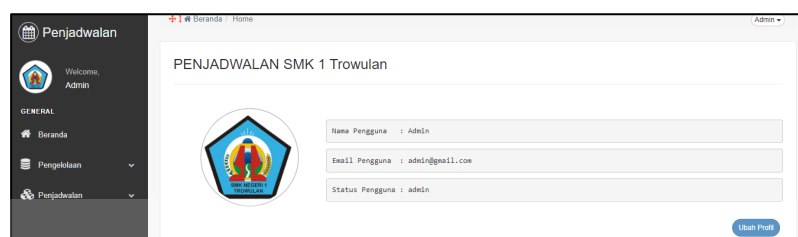
4.1.2 Implementasi Sistem

Berikut ini merupakan Aplikasi penjadwalan SMK 1 Trowulan. Untuk lebih detailnya dapat dilihat pada Lampiran 8.

A. Admin

A.1. Tampilan Halaman Beranda

Berikut ini merupakan tampilan halaman beranda pada admin dimana berisi akun yang dapat diubah oleh admin.



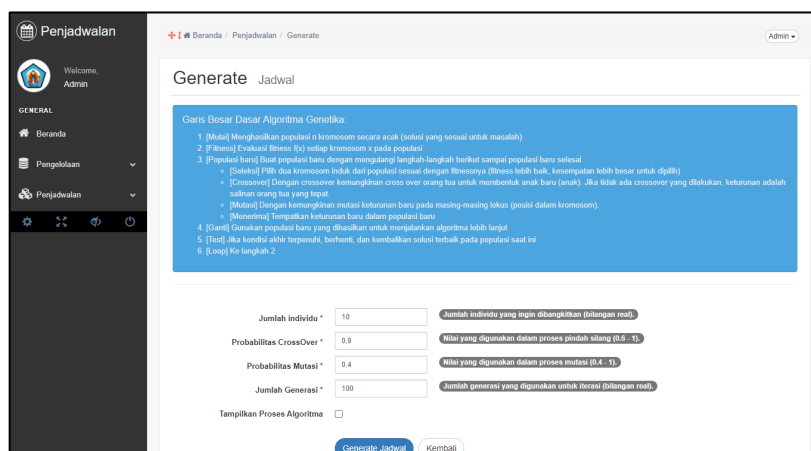
Gambar 4.1. Tampilan Halaman Beranda

A.2. Tampilan Halaman Buat Jadwal

Pada tampilan ini digunakan untuk menambahkan jadwal dengan mengisi beberapa *field* komponen algoritma genetika.

1. Tampilan Halaman Awal Buat Jadwal (Hanya Untuk Ujian)

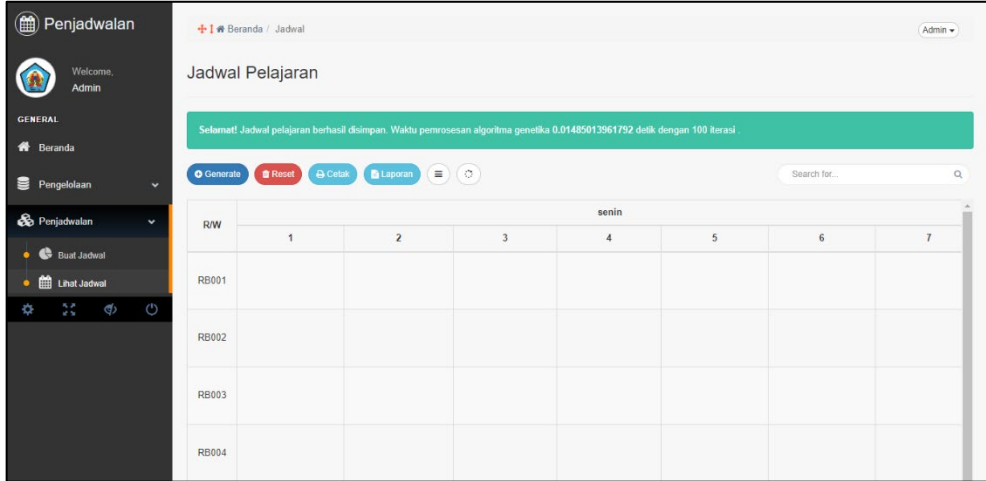
Pada tampilan awal berisi form untuk membuat jadwal, dengan mengisi *field* algoritma lalu dapat di *generate* jadwal.



Gambar 4.2. Tampilan Halaman Awal Buat Jadwal (Untuk Ujian)

2. Tampilan Halaman Berhasil *Generate*

Tampilan berikut ini berisi jadwal yang telah di *generate* dari algoritma genetika.



The screenshot shows the 'Penjadwalan' application interface. A green banner at the top states: 'Selamat! Jadwal pelajaran berhasil disimpan. Waktu pemrosesan algoritma genetika 0.01485013961792 detik dengan 100 iterasi.' Below the banner are buttons for 'Generate', 'Reset', 'Cetak', and 'Laporan'. The main area displays a table titled 'Jadwal Pelajaran' for 'senin' (Monday). The table has columns for days (1-7) and rows for room numbers (RW). The table is currently empty.

RW	senin						
	1	2	3	4	5	6	7
RB001							
RB002							
RB003							
RB004							

Gambar 4.3. Tampilan Halaman Berhasil Generate

B. Siswa, Guru, dan Kepsek

Pada akses berikut ini siswa, guru, kepek hanya dapat melihat semua jadwal yang telah dibuat oleh admin. Selain itu jadwal yang telah dibuat dapat dicetak atau dikelompokkan sesuai jurusan.



The screenshot shows the 'Penjadwalan' application interface for a user named 'Siswa'. The main area displays a detailed lesson schedule for Monday. The table has columns for days (1-7) and rows for room numbers (RW). The table is filled with lesson details, including subject names, instructor names, and time slots.

RB009							
RB010	IPS - Ekonomi (Peminatan) Eni Purwantini senin, 08:00:00-09:30:00	IPS - Ekonomi (Peminatan) Dwi Risyanto, S.Pd. senin, 09:45:00-11:15:00					
RB011	IPS - Pendidikan Agama Hindu Mira Khoirunisa, S.pd. senin, 08:00:00-09:30:00	IPS - Pendidikan Agama Kristen Drs. Arum Triharjana senin, 09:45:00-11:15:00	IPS - Pendidikan Agama Kristen Drs. Sukur senin, 11:15:00-12:00:00	IPS - Pendidikan Agama Kristen Dra. Ngadyana senin, 12:30:00-14:00:00	IPS - Pendidikan Agama Kristen Sri Suharti, S.Pd. senin, 12:30:00-13:15:00	IPS - Bhs & Sastra Inggris Ambar Pratitis, S.Pd. senin, 14:00:00-15:30:00	IPS - Fisika (Lintas Minat) Efi Triana Ningrum S.Pd. senin, 13:15:00-14:00:00
RB012	IPA - Bhs & Sastra Inggris Drs. Sukur senin, 08:00:00-09:30:00	IPA - Pendidikan Agama Kristen Yulianti Prithdyastuti, S.Pd. senin, 09:45:00-11:15:00	IPA - Pendidikan Agama Kristen Dra. Veni Pro Deo senin, 11:15:00-12:00:00	IPA - Bhs & Sastra Inggris Dra. Sri Maesarini Ki senin, 12:30:00-14:00:00	IPA - Fisika (Peminatan) Drs. Susiyanta senin, 12:30:00-13:15:00	IPA - Fisika (Peminatan) Febryardini Dian P.R, S.S. senin, 14:00:00-15:30:00	IPA - Matematika (Peminatan) Nini Kurniawati, S.Pd. senin, 13:15:00-14:00:00

Gambar 4.4. Tampilan Jadwal Pada User Siswa, Guru, Kepsek

4.1.3 Hasil Uji Coba Sistem


Serangkaian uji coba telah dilakukan untuk menguji efektivitas dan kehandalan aplikasi penjadwalan di SMK 1 Trowulan. Hasil-hasil uji coba tersebut disajikan secara rinci dalam Lampiran 9, memberikan pemahaman mendalam mengenai kinerja dan fungsionalitas aplikasi tersebut. Memproses Data Penjadwalan

Proses pengolahan data penjadwalan dimulai dengan memasukkan informasi mengenai 12 ruangan, 47 guru, pelajaran kelas 10, kelas 11, kelas 12, dan 36 waktu ujian yang diperlukan. Data ini menjadi landasan penting saat melakukan pembuatan jadwal ujian.

1. Penentuan Parameter GA

Untuk meningkatkan efektivitas aplikasi penjadwalan, diperlukan beberapa percobaan untuk menentukan parameter atau input yang optimal. Percobaan dilakukan untuk mengoptimalkan kinerja aplikasi penjadwalan dengan mengatur beberapa parameter kunci. Pertama, pengujian pengaruh variasi jumlah individu dalam populasi. Sebanyak 3 individu digunakan sebagai basis, dan hasilnya dibandingkan dengan percobaan menggunakan 1 individu dan 2 individu.

Dalam Gambar 4.5 ditemukan bahwa penggunaan 3 individu menghasilkan jadwal pada generasi yang lebih rendah dari pada menggunakan 2 atau 1 individu yang berarti semakin banyak individu akan lebih baik. Selanjutnya, percobaan kedua untuk probabilitas *crossover* sebagai parameter yang akan diuji. Dengan probabilitas *crossover* pertama sebesar 0.6, kedua probabilitas *crossover* lebih rendah (0.3) dan ketiga dengan probabilitas lebih tinggi (0.8).



Parameter	
[Jumlah Individu]	: 1
[Probabilitas Crossover]	: 0.6
[Probabilitas Mutasi]	: 0.1
[Jumlah Generasi]	: 100
Waktu Yang Di Butuhkan: 0.25592589378357 detik	
Memory Usage: 3336 kilo bytes	
GENERASI: 4	
Parameter	
[Jumlah Individu]	: 2
[Probabilitas Crossover]	: 0.6
[Probabilitas Mutasi]	: 0.1
[Jumlah Generasi]	: 100
Waktu Yang Di Butuhkan: 0.41406512260437 detik	
Memory Usage: 3521.3203125 kilo bytes	
GENERASI: 3	
Parameter	
[Jumlah Individu]	: 3
[Probabilitas Crossover]	: 0.6
[Probabilitas Mutasi]	: 0.1
[Jumlah Generasi]	: 100
Waktu Yang Di Butuhkan: 0.42133378982544 detik	
Memory Usage: 3615.1953125 kilo bytes	
GENERASI: 2	

Gambar 4.5. Uji Coba Jumlah Individu

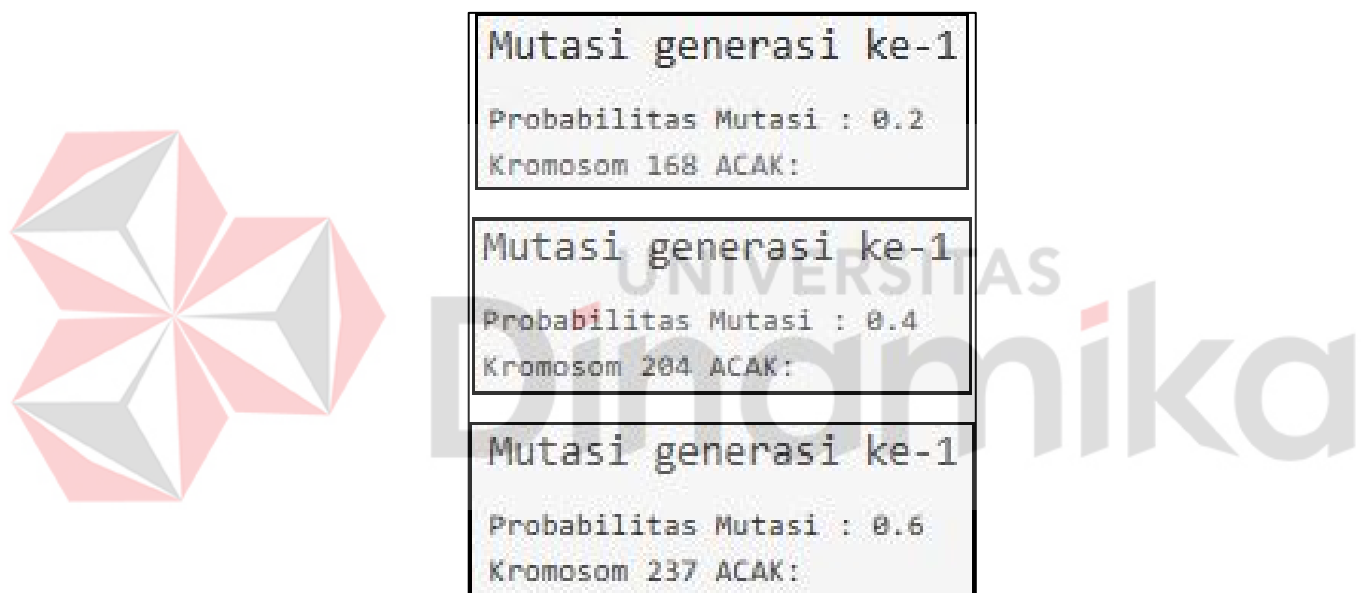
Dari hasil percobaan pada gambar 4.6 dapat disimpulkan bahwa probabilitas crossover 0.8 lebih baik dalam menentukan individu yang akan dijadikan induk karena probabilitas untuk mendapatkan induk yang cocok lebih besar, yang menghasilkan jadwal yang lebih optimal dibandingkan dengan nilai probabilitas *crossover* lainnya. Selanjutnya, gambar tersebut menampilkan hasil dari percobaan pada probabilitas mutasi. Dengan probabilitas mutasi awal sebesar 0.4, dilakukan pengujian dengan nilai probabilitas yang lebih rendah (0.2) dan lebih tinggi (0.6).



<p>BILANGAN ACAK: Random[0] : 0.32412442486925 Random[1] : 0.98634659870823 Random[2] : 0.5393360622876</p> <p>KROMOSOM INDUK:</p> <p>Crossover Rate: 0.3</p> <p>PROSES CROSSOVER:</p>
<p>BILANGAN ACAK: Random[0] : 0.20741836876023 Random[1] : 0.17701303408342 Random[2] : 0.28840370396543</p> <p>KROMOSOM INDUK:</p> <p>Crossover Rate: 0.6 Parent[0] : 0 Parent[1] : 1 Parent[2] : 2</p>
<p>BILANGAN ACAK: Random[0] : 0.10751978685498 Random[1] : 0.81050560521451 Random[2] : 0.41103485245771</p> <p>KROMOSOM INDUK:</p> <p>Crossover Rate: 0.8 Parent[0] : 0 Parent[1] : 2</p>

Gambar 4.6. Uji Coba Probabilitas Crossover

Pada gambar 4.7, terlihat bahwa probabilitas mutasi sebesar 0.6 menghasilkan variasi populasi yang lebih besar dan solusi yang lebih unggul dibandingkan dengan nilai probabilitas mutasi lainnya. Selanjutnya, gambar tersebut menampilkan hasil dari percobaan pada probabilitas mutasi. Dengan probabilitas mutasi awal sebesar 0.4, dilakukan pengujian dengan nilai probabilitas yang lebih rendah (0.2) dan lebih tinggi (0.6). Dari gambar pada gambar 4.8, terlihat bahwa probabilitas mutasi sebesar 0.6 menghasilkan variasi populasi yang lebih besar dan solusi yang lebih unggul dibandingkan dengan nilai probabilitas mutasi lainnya.



Gambar 4.7. Uji Coba Probabilitas Mutasi

Selanjutnya pengguna siap untuk mengisi beberapa *field* komponen. Di antaranya, pengguna dapat memasukkan jumlah individu, probabilitas *crossover*, probabilitas mutasi, dan jumlah generasi. Jumlah individu menentukan berapa banyak individu yang akan dimasukkan dalam populasi. Probabilitas *crossover* digunakan untuk mengendalikan frekuensi operator *crossover*, sedangkan probabilitas mutasi digunakan untuk meningkatkan variasi populasi. Jumlah generasi menentukan berapa kali iterasi akan dilakukan. Dalam pengujian kali ini,

input yang diberikan adalah 3 individu, probabilitas *crossover* 0.8, probabilitas mutasi 0.6, dan jumlah generasi 100.

Jumlah individu *	3	Jumlah individu yang ingin dibangkitkan (bilangan real).
Probabilitas CrossOver *	0,8	Nilai yang digunakan dalam proses pindah silang (0.1 - 1).
Probabilitas Mutasi *	0,6	Nilai yang digunakan dalam proses mutasi (0.1 - 1).
Jumlah Generasi *	100	Jumlah generasi yang digunakan untuk iterasi (bilangan real).

Gambar 4.8. Penentuan Parameter GA







2. Populasi awal

Pada populasi awal akan membuat 3 individu dengan kromosom yang berbeda-beda untuk setiap populasi. Setiap individu akan dihasilkan dengan memilih kelas secara acak dari daftar kelas yang tersedia, memilih ruangan yang tersedia secara acak, memilih waktu yang tersedia secara acak, dan memilih guru yang tersedia secara acak dengan memperhatikan pengecualian terhadap waktu guru yang sudah terdaftar dalam daftar *request* waktu dimana guru tersebut tidak dapat menjadi pengawas pada waktu yang dapat dilihat pada gambar 4.9 terlihat bahwa terdapat daftar permintaan waktu dari Ibu Eni yang menunjukkan bahwa pada hari Jumat, Bu Eni tidak dapat menjadi pengawas.

Guru *	Eni Purwantini		
Waktu *	Pilih waktu		
Jam ke-	Hari	Jam	Aksi
33	jumat	07:00:00 - 08:00:00	Hapus
34	jumat	08:00:00 - 09:00:00	Hapus
35	jumat	09:15:00 - 10:15:00	Hapus
36	jumat	10:15:00 - 11:15:00	Hapus

Gambar 4.9. Request Waktu Satu

Pada gambar 4.10 terlihat bahwa terdapat daftar permintaan waktu dari Ibu Eni yang menunjukkan bahwa pada hari Jumat, Bapak Prangripta tidak dapat menjadi pengawas.

Guru *	Prangripta Wibawa		
Waktu *	Pilih waktu		
Jam ke-	Hari	Jam	Aksi
17	rabu	07:00:00 - 08:00:00	
18	rabu	08:00:00 - 09:00:00	
20	rabu	10:15:00 - 11:15:00	
21	rabu	11:30:00 - 12:30:00	
23	rabu	14:00:00 - 15:00:00	
24	rabu	15:00:00 - 16:00:00	

Gambar 4.10. Request Waktu Kedua

Informasi ini kemudian disusun menjadi kromosom – kromosom dalam individu – individu yang dapat dilihat pada gambar 4.11.

Generasi ke-1	
POPULASI AWAL:	
Individu[0]:	[001, 001, 001, 047], [006, 002, 001, 047],
Individu[1]:	[001, 001, 001, 025], [006, 002, 001, 008],
Individu[2]:	[001, 001, 001, 047], [006, 002, 001, 032],

Gambar 4.11. Populasi Awal

3. Hitung Nilai *Fitness*

Setelah populasi awal dihasilkan, langkah selanjutnya adalah mengevaluasi setiap individu dalam populasi berdasarkan kriteria yang ditentukan. Dalam konteks penjadwalan ujian, kriteria dapat mencakup preferensi waktu, kecocokan

kelas dengan ruangan yang dipilih, dan kesesuaian jadwal dengan ketersediaan guru / pengawas. Penjelasan perhitungan nilai *fitness* berdasarkan kriteria yang disebutkan adalah sebagai berikut:

a. Banyaknya Pelanggaran Pengawas di Waktu yang Sama:

Pelanggaran ini terjadi ketika ada dua ujian atau lebih yang dijadwalkan pada waktu yang sama dan diawasi oleh pengawas yang sama. Jumlah pelanggaran pengawas di waktu yang sama dapat dihitung dengan mengidentifikasi jumlah pasangan ujian yang dijadwalkan pada waktu yang sama dan diawasi oleh pengawas yang sama. Semakin rendah jumlah pelanggaran pengawas di waktu yang sama, semakin baik nilai *fitness*nya.

b. Banyaknya Pelanggaran Ruangan di Waktu yang Sama:

Pelanggaran ini terjadi ketika ada dua ujian atau lebih yang dijadwalkan pada waktu yang sama dan menggunakan ruangan yang sama. Jumlah pelanggaran ruangan di waktu yang sama dapat dihitung dengan mengidentifikasi jumlah pasangan ujian yang dijadwalkan pada waktu yang sama dan menggunakan ruangan yang sama. Jumlah pelanggaran ruangan di waktu yang sama dapat dihitung dengan mengidentifikasi jumlah pasangan ujian yang dijadwalkan pada waktu yang sama dan menggunakan ruangan yang sama.

c. Banyaknya Pelanggaran Mata Pelajaran di Waktu yang Sama:

Pelanggaran ini terjadi ketika ada dua ujian atau lebih yang dijadwalkan pada waktu yang sama dan memiliki mata pelajaran yang sama. Jumlah pelanggaran mata pelajaran di waktu yang sama dapat dihitung dengan mengidentifikasi jumlah pasangan ujian yang dijadwalkan pada waktu yang sama dan memiliki mata pelajaran yang sama. Semakin rendah jumlah pelanggaran mata pelajaran di waktu

yang sama, semakin baik nilai *fitness*nya. Setelah menghitung jumlah pelanggaran pengawas, ruangan, dan mata pelajaran di waktu yang sama, langkah selanjutnya adalah menghitung nilai *fitness* setiap individu menggunakan formula berikut:

$$F[\text{individu}] = 1 / (1 + \text{Jumlah Pelanggaran Mata Pelajaran} + \text{Jumlah Pelanggaran Pengawas} + \text{Jumlah Pelanggaran Ruangan}).$$

Dalam rumus diatas, semakin banyak pelanggaran yang terjadi, nilai *fitness* akan semakin rendah. Sebaliknya, semakin sedikit pelanggaran yang terjadi, nilai *fitness* akan semakin tinggi. Pada gambar 4.12 terdapat perhitungan nilai kebugaran (*fitness*) untuk tiga individu dalam populasi yang diamati, yang direpresentasikan sebagai F[0], F[1], dan F[2]. Berikut adalah penjelasan mengenai perhitungan dan hasilnya:

a. F[0]:

- 1) Jumlah Pelanggaran Pengawas = 44 (ada 44 pasangan ujian yang menggunakan pengawas yang sama pada waktu yang sama).
- 2) Jumlah Pelanggaran Ruangan = 0 (tidak ada ruangan ruangan di waktu yang sama).
- 3) Jumlah Pelanggaran Mata Pelajaran = 0 (tidak ada pelanggaran mata pelajaran di waktu yang sama).
- 4) Kemudian dimasukkan kedalam formula $f = 1 / (1 + 0 + 44 + 0)$ dan menghasilkan nilai *fitness* untuk individu *index* ke 0 yaitu 0.0222222222222222.

b. F[1]:

- 1) Jumlah Pelanggaran Pengawas = 26 (ada 26 pasangan ujian yang menggunakan pengawas yang sama pada waktu yang sama).

- 2) Jumlah Pelanggaran Ruang = 0 (tidak ada ruangan ruangan di waktu yang sama).
- 3) Jumlah Pelanggaran Mata Pelajaran = 0 (tidak ada pelanggaran mata pelajaran di waktu yang sama).
- 4) Kemudian dimasukkan kedalam formula $f = 1 / (1 + 0 + 26 + 0)$ dan menghasilkan nilai *fitness* untuk individu *index* ke 1 yaitu 0.037037037037037.

c. F[2]:

- 1) Jumlah Pelanggaran Pengawas = 36 (ada 36 pasangan ujian yang menggunakan pengawas yang sama pada waktu yang sama).
- 2) Jumlah Pelanggaran Ruang = 0 (tidak ada ruangan ruangan di waktu yang sama).
- 3) Jumlah Pelanggaran Mata Pelajaran = 0 (tidak ada pelanggaran mata pelajaran di waktu yang sama).
- 4) Kemudian dimasukkan kedalam formula $f = 1 / (1 + 0 + 36 + 0)$ dan menghasilkan nilai *fitness* untuk individu *index* ke 2 yaitu 0.027027027027027.

Setelah mendapatkan semua nilai *fitness* setiap individu maka dapat menghitung total nilai *fitness* (F) dengan menjumlahkan nilai *fitness* dari setiap individu dalam populasi.

$$\text{Total } F = F[0] + F[1] + F[2] = 0.022222222222222 + 0.037037037037037 + 0.027027027027027 = 0.086286286286286$$

Karena tidak ditemukan nilai *fitness* 1 dalam setiap individu dalam populasi, proses algoritma genetika akan dilanjutkan ke tahap selanjutnya proses seleksi.

```

NILAI FITNESS:
F[0]: 1/(1+0+44+0) = 0.0222222222222222
F[1]: 1/(1+0+26+0) = 0.037037037037037
F[2]: 1/(1+0+36+0) = 0.027027027027027
Total F: 0.086286286286286

```

Gambar 4.12. Hitung Nilai *Fitness*

4. Seleksi

Proses seleksi yaitu proses pemilihan kromosom untuk dijadikan induk. Berikut adalah penjelasan perhitungan pada gambar 4.13 pada proses algoritma genetika probabilitas sampai probabilitas kumulatif dengan jumlah 3 individu:

a. Probabilitas

Probabilitas dihitung berdasarkan skor *fitness* setiap individu. Skor *fitness* individu dibagi dengan total skor *fitness* semua individu untuk mendapatkan probabilitas relatif. Proses sebelumnya individu[0] memiliki skor *fitness* 0.022222222222222, individu[1] memiliki skor *fitness* 0.037037037037037, dan individu[2] memiliki skor *fitness* 0.027027027027027. Total skor *fitness* adalah $0.022222222222222 + 0.037037037037037 + 0.027027027027027 = 0.086286286286286$.

$$1) \text{ Probabilitas individu}[0] = 0.022222222222222 / 0.086286286286286 =$$

$$0,38461538461538$$

$$2) \text{ Probabilitas individu}[1] = 0.037037037037037 / 0.086286286286286 =$$

$$0.23076923076923$$

$$3) \text{ Probabilitas individu}[2] = 0.027027027027027 / 0.086286286286286 =$$

$$0,38461538461538$$

b. Probabilitas Kumulatif

Probabilitas kumulatif adalah probabilitas individu terpilih dari individu pertama hingga individu saat ini dalam urutan terurut menurun berdasarkan probabilitas seleksi mereka. Probabilitas kumulatif dihitung dengan menjumlahkan probabilitas seleksi dari individu pertama hingga individu ke tiga.

- 1) Probabilitas kumulatif individu[0] = 0.25754060324826
- 2) Probabilitas kumulatif individu[0] = 0.25754060324826 + 0.4292343387471
= 0.68677494199536
- 3) Probabilitas kumulatif individu[0] = 0.68677494199536 + 0.31322505800464
= 1



```

PROBABILITAS :
P[0] : 0.25754060324826
P[1] : 0.4292343387471
P[2] : 0.31322505800464
Total P: 1

PROBABILITAS KUMULATIF :
PK[0] : 0.25754060324826
PK[1] : 0.68677494199536
PK[2] : 1

```

Gambar 4.13. Proses Probabilitas Kumulatif

Setelah menghitung probabilitas akumulatif untuk setiap individu. Selanjutnya memilih individu-individu berdasarkan angka acak yang telah dibangkitkan. Jika angka acak tersebut lebih kecil dari probabilitas akumulatif individu pertama, individu pertama dipilih. Jika angka acak tersebut berada di antara probabilitas akumulatif individu pertama dan kedua, individu kedua dipilih. Jika angka acak tersebut lebih besar dari probabilitas akumulatif individu kedua, individu ketiga dipilih.

Berikut adalah penjelasan perhitungan pada gambar 4.14 pada proses seleksi algoritma genetika dengan jumlah 3 individu pada literasi ke 1:

a. Angka $Random[0] = 0.81560848132503$

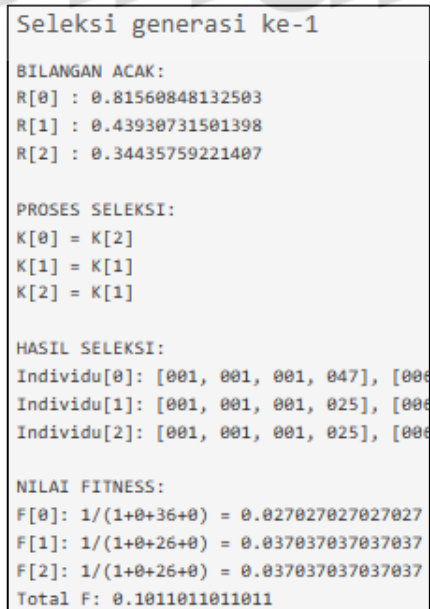
Karena angka *random index* ke 0 berada di antara $K[1]$ dengan $K[2]$ maka individu[0] baru di isi dengan individu[2] pada proses populasi awal generasi pertama.

b. Angka $Random[1] = 0.43930731501398$

Karena angka *random index* ke 1 berada di antara $K[0]$ dengan $K[1]$ maka individu[1] baru di isi dengan individu[1] pada proses populasi awal generasi pertama.

c. Angka $Random[2] = 0.34435759221407$

Karena angka *random index* ke 2 berada di antara $K[0]$ dengan $K[1]$ maka individu[2] baru di isi dengan individu[1] pada proses populasi awal generasi pertama.



```

Seleksi generasi ke-1

BILANGAN ACAK:
R[0] : 0.81560848132503
R[1] : 0.43930731501398
R[2] : 0.34435759221407

PROSES SELEKSI:
K[0] = K[2]
K[1] = K[1]
K[2] = K[1]

HASIL SELEKSI:
Individu[0]: [001, 001, 001, 047], [000
Individu[1]: [001, 001, 001, 025], [000
Individu[2]: [001, 001, 001, 025], [000

NILAI FITNESS:
F[0]: 1/(1+0+36+0) = 0.027027027027027
F[1]: 1/(1+0+26+0) = 0.037037037037037
F[2]: 1/(1+0+26+0) = 0.037037037037037
Total F: 0.1011011011011

```

Gambar 4.14. Proses Seleksi

5. Crossover

Crossover, yaitu proses penyilangan kromosom sehingga membentuk kromosom baru (*offspring*) yang harapannya lebih baik dari pada kualitas induknya. Pada gambar 4.15 terlihat langkah-langkah proses *crossover* yang dilakukan. Pada awalnya, sistem melakukan pengacakan bilangan acak sebanyak tiga kali dengan rentang nilai antara 0 hingga 1. Hasil pengacakan menunjukkan bahwa *random*[0] memiliki nilai 0.45, *random*[1] memiliki nilai 0.64, dan *random*[2] memiliki nilai 0.26. Selanjutnya, nilai-nilai acak tersebut dibandingkan dengan tingkat persilangan (*crossover rate*) yang telah diinputkan sebelumnya, yaitu 0.6.

Dalam proses pertama, *individu*[0] dan *individu*[2] dipilih sebagai induk untuk dikawinkan karena nilai acak *random*[0] dan *random*[2] lebih kecil daripada nilai *crossover rate*. Dengan demikian, *individu*[0] menjadi induk pertama yang dikawinkan dengan *individu*[2], dan sebaliknya, *individu*[2] dikawinkan dengan *individu*[0]. Selanjutnya, perlu ditentukan titik potong (*one cut-point crossover*) untuk proses *crossover*. Penentuan titik potong dilakukan dengan menghasilkan bilangan acak antara 1 hingga (panjang kromosom - 1) sebanyak jumlah *parent* yang terlibat, yaitu 2. Dalam proses *crossover* pada generasi ke 1, setelah melakukan randomisasi, diperoleh hasil sebagai berikut: titik potong pertama, yaitu 24, digunakan untuk pasangan *individu*[0] dengan *individu*[1]. Sedangkan titik potong kedua, yaitu 105, digunakan untuk pasangan *individu*[1] dengan *individu*[2]. Sedangkan titik potong ketiga, yaitu 207, digunakan untuk pasangan *individu*[2] dengan *individu*[0]. Titik potong tersebut akan membagi kromosom-kromosom *parent* menjadi dua bagian, di mana bagian sebelum titik potong akan diambil dari salah satu *parent*, dan bagian setelah titik potong diambil dari *parent* lainnya.

Proses ini memungkinkan pertukaran informasi gen antara kedua *parent* yang berbeda dan memperkenalkan variasi genetik dalam populasi. Selanjutnya, kualitas *offspring* ditentukan melalui evaluasi menggunakan fungsi *fitness* yang telah ditetapkan sebelumnya, yang mengukur kelayakan solusi jadwal yang dihasilkan oleh *offspring*.

Setelah proses *crossover*, kualitas *offspring* ditentukan melalui evaluasi menggunakan fungsi *fitness* yang telah ditetapkan sebelumnya. Fungsi *fitness* ini memberikan nilai berdasarkan kelayakan solusi jadwal yang dihasilkan oleh *offspring*. Karena nilai *fitness* belum mencapai target yang ditentukan yaitu nilai *fitness* 1, maka proses selanjutnya dilakukan proses mutasi.



```

Pindah silang generasi ke-1

BILANGAN ACAK:
Random[0] : 0.22885966078791
Random[1] : 0.6709179280656
Random[2] : 0.12333343602872

KROMOSOM INDUK:
Crossover Rate: 0.8
Parent[0] : 0
Parent[1] : 1
Parent[2] : 2

PROSES CROSSOVER:
Crossover ke 0: 0 x 1 dengan offspring: 24
Individu 0 : [001, 001, 001, 047],
Individu 1 : [001, 001, 001, 025],
Hasil : [001, 001, 001, 047],
Crossover ke 1: 1 x 2 dengan offspring: 105
Individu 1 : [001, 001, 001, 025],
Individu 2 : [001, 001, 001, 025],
Hasil : [001, 001, 001, 025],
Crossover ke 2: 2 x 0 dengan offspring: 207
Individu 2 : [001, 001, 001, 025],
Individu 0 : [001, 001, 001, 047],
Hasil : [001, 001, 001, 025],

HASIL CROSSOVER:
Individu[0]: [001, 001, 001, 047], [006, 002, 0
Individu[1]: [001, 001, 001, 025], [006, 002, 0
Individu[2]: [001, 001, 001, 025], [006, 002, 0


NILAI FITNESS:
F[0]: 1/(1+0+28+0) = 0.03448275862069
F[1]: 1/(1+0+26+0) = 0.037037037037037
F[2]: 1/(1+0+26+0) = 0.037037037037037
Total F: 0.10855683269476

```

Gambar 4.15. Crossover

6. Mutasi

Mutasi, yaitu proses perubahan nilai dari satu atau beberapa gen pada kromosom. Pada gambar 4.15 terdapat 3 individu dalam populasi yang dihasilkan setelah proses gen acak pada tahap mutasi pada algoritma genetika. Pada Gambar 4.16, diterapkan proses perubahan (*mutation*) gen acak yang akan mengubah 60% dari kromosom-kromosom yang ada pada tiga individu yang diamati.



```

Mutasi generasi ke-1

Probabilitas Mutasi : 0.6
Kromosom 251 ACAK:
103 : [2, 30]
64  : [1, 27]
4   : [0, 3]
101 : [2, 28]
101 : [2, 28]
5   : [0, 4]
13  : [0, 12]
53  : [1, 16]
22  : [0, 21]

```

Gambar 4.16. Mutasi Generasi Acak

Pada gambar 4.17 dapat dilihat setelah selesai perubahan 60% dari jumlah kromosom pada semua individu akan dilakukan evaluasi dengan menggunakan fungsi *fitness*.

```

HASIL MUTASI:
Individu[0]: [001, 001, 001, 013], [006, 002, 001, 032],
Individu[1]: [001, 001, 001, 007], [006, 002, 001, 008],
Individu[2]: [001, 001, 001, 025], [006, 002, 001, 008],

NILAI FITNESS:
F[0]: 1/(1+0+14+0) = 0.0666666666666667
F[1]: 1/(1+0+10+0) = 0.09090909090909091
F[2]: 1/(1+0+12+0) = 0.076923076923077
Total F: 0.23449883449883

```

Gambar 4.17. Hasil Mutasi

7. Perolehan Jadwal

Pada Gambar 4.18, diterapkan proses pengubahan (mutation) gen acak yang akan mengubah 60% dari kromosom-kromosom yang ada pada tiga individu yang diamati.

```

Mutasi generasi ke-2
Probabilitas Mutasi : 0.6
Kromosom 147 ACAK:
79 : [2, 6]
97 : [2, 24]
100 : [2, 27]
31 : [0, 30]
29 : [0, 28]

```

Gambar 4.18. Gen Acak Generasi 2

Gambar 4.19 merupakan hasil kromosom terbaik setelah generasi ke 2 dan memiliki nilai *fitness* 1 pada individu *index* ke 0.

```

HASIL MUTASI:
Individu[0]: [001, 001, 001, 024], [006, 002, 001, 008],
Individu[1]: [001, 001, 001, 013], [006, 002, 001, 032],
Individu[2]: [001, 001, 001, 018], [006, 002, 001, 014],

NILAI FITNESS:
F[0]: 1/(1+0+0+0) = 1
F[1]: 1/(1+0+4+0) = 0.2
F[2]: 1/(1+0+8+0) = 0.111111111111111
Total F: 1.31111111111111

```

Gambar 4.19. Hasil Perolehan Mutasi Generasi 2

8. Perolehan Jadwal UTS / UAS

Berikut merupakan hasil jadwal setelah *generate* jadwal selesai. diperoleh hasil jadwal yang terdiri dari 240 jadwal untuk 7 jurusan yang ada di Sekolah

Menengah Kejuruan (SMK). Jadwal ini mencakup periode waktu dari hari Senin hingga Jumat. Selain itu, jadwal ini juga mempertimbangkan ketersediaan 47 pengawas yang terlibat dalam proses pengajaran di SMK 1 Trowulan.

Hasil jadwal yang dihasilkan oleh metode Algoritma Genetika menunjukkan kemampuannya dalam menyusun penjadwalan yang efisien dan memenuhi kriteria yang telah ditetapkan sebelumnya. Jadwal ini menghindari tumpang tindih waktu antara mata pelajaran, mempertimbangkan preferensi pegawai, serta meminimalkan konflik jadwal yang mungkin terjadi.

R/W	senin			
	senin, 07:00:00 - 08:00:00	senin, 08:00:00 - 09:00:00	senin, 09:15:00 - 10:15:00	senin, 10:15:00 - 11:15:00
RB001	KKLP - Bahasa Inggris Drs. Gundari Yanto, M.MPd. senin, 07:00:00-08:00:00 Ubah	KKLP - Bahasa Jawa Dra. Wisnandari senin, 08:00:00-09:00:00 Ubah	KKLP - Pendidikan Pancasila dan Kewarganegaraan Dra. Sri Netty Purwaningsih senin, 09:15:00-10:15:00 Ubah	KKLP - Matematika Rofiu Inayah S.Pd senin, 10:15:00- 11:15:00 Ubah
RB002	DKV - Bahasa Inggris Yoni Rakhmat Yunus, S.Pd., M.MP senin, 07:00:00-08:00:00 Ubah	DKV - Bahasa Jawa Drs. Mulyak senin, 08:00:00-09:00:00 Ubah	DKV - Pendidikan Pancasila dan Kewarganegaraan Dra. Veni Pro Deo senin, 09:15:00-10:15:00 Ubah	DKV - Matematika Eni Purwantini senin, 10:15:00- 11:15:00 Ubah

Gambar 4.20. Hasil Jadwal Ujian

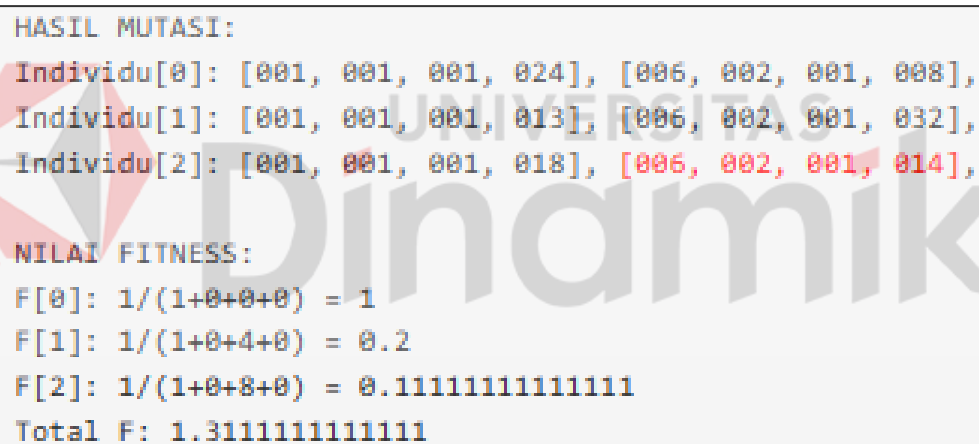
4.2. Evaluasi Sistem

Setelah melakukan implementasi sistem penjadwalan untuk SMK 1 Trowulan, dan melaksanakan serangkaian pengujian untuk menguji coba sistem. Dalam hasil uji coba ini, terdapat beberapa aspek yang diobservasi secara mendalam, termasuk *crash* jadwal dan waktu yang dibutuhkan dalam proses

penyusunan jadwal menggunakan algoritma genetika. Berikut ini adalah beberapa temuan yang signifikan:

1. *Crash* jadwal

Pada tahap pengujian, ditemukan beberapa situasi di mana jadwal pada sistem mengalami *crash* yang dapat terlihat dengan jelas melalui penandaan angka berwarna merah. Salah satu temuan penting adalah adanya situasi di mana beberapa individu (jadwal) mengalami bentrokan, dimana terdapat jadwal yang saling tumpang tindih atau bertabrakan. Pada individu 0, terdapat temuan yang menarik, di mana jadwal-jadwal pada individu tersebut terindikasi tidak mengalami bentrokan atau tumpang tindih.



```
HASIL MUTASI:
Individu[0]: [001, 001, 001, 024], [006, 002, 001, 008],
Individu[1]: [001, 001, 001, 013], [006, 002, 001, 032],
Individu[2]: [001, 001, 001, 018], [006, 002, 001, 014],
NILAI FITNESS:
F[0]: 1/(1+0+0+0) = 1
F[1]: 1/(1+0+4+0) = 0.2
F[2]: 1/(1+0+8+0) = 0.111111111111111
Total F: 1.31111111111111
```

Gambar 4.21. Contoh Crash Jadwal

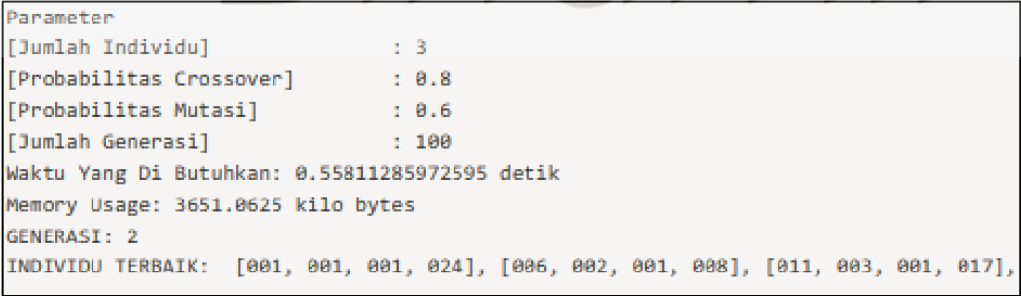
2. Mempercepat waktu pembuatan jadwal ujian

Saat proses manual dalam penyusunan jadwal ujian memakan waktu 1 hingga 2 minggu setelah semua data terkumpul. Namun, dengan sistem penjadwalan, durasi eksekusi yang sangat cepat, sekitar 0.558 detik, menunjukkan kemampuan sistem dalam mengolah dan menyusun jadwal ujian secara cepat. Dalam konteks penyelesaian masalah optimisasi yang kompleks, durasi tersebut mencerminkan efektivitas algoritma genetika yang digunakan oleh sistem.

Dengan kecepatan eksekusi yang sekecil itu, sistem penjadwalan memberikan manfaat yang signifikan bagi SMK 1 Trowulan. Proses manual yang dulu memakan waktu semingguan sekarang dapat dihindari, dan penyusunan jadwal dapat dilakukan secara *real-time* dengan hasil yang optimal.

Kecepatan eksekusi yang tinggi juga berarti sistem ini dapat dengan mudah mengelola jumlah data yang besar dalam waktu singkat, memberikan fleksibilitas dan skalabilitas yang penting bagi lembaga dengan tingkat kepadatan jadwal yang tinggi. Dengan efisiensi yang tinggi, tenaga kerja manusia dapat dialokasikan untuk tugas-tugas lain yang memerlukan keterlibatan lebih mendalam.

Kesimpulannya, pergeseran dari pendekatan manual yang memakan waktu menjadi penggunaan sistem penjadwalan dengan waktu eksekusi sekitar 0.558 detik menegaskan kemampuan sistem dalam mengoptimalkan penyusunan jadwal ujian dengan cepat, memberikan dampak positif pada efisiensi keseluruhan dan produktivitas SMK 1 Trowulan.



```

Parameter
[Jumlah Individu]           : 3
[Probabilitas Crossover]   : 0.8
[Probabilitas Mutasi]      : 0.6
[Jumlah Generasi]          : 100
Waktu Yang Di Butuhkan: 0.55811285972595 detik
Memory Usage: 3651.0625 kilo bytes
GENERASI: 2
INDIVIDU TERBAIK: [001, 001, 001, 024], [006, 002, 001, 008], [011, 003, 001, 017],

```

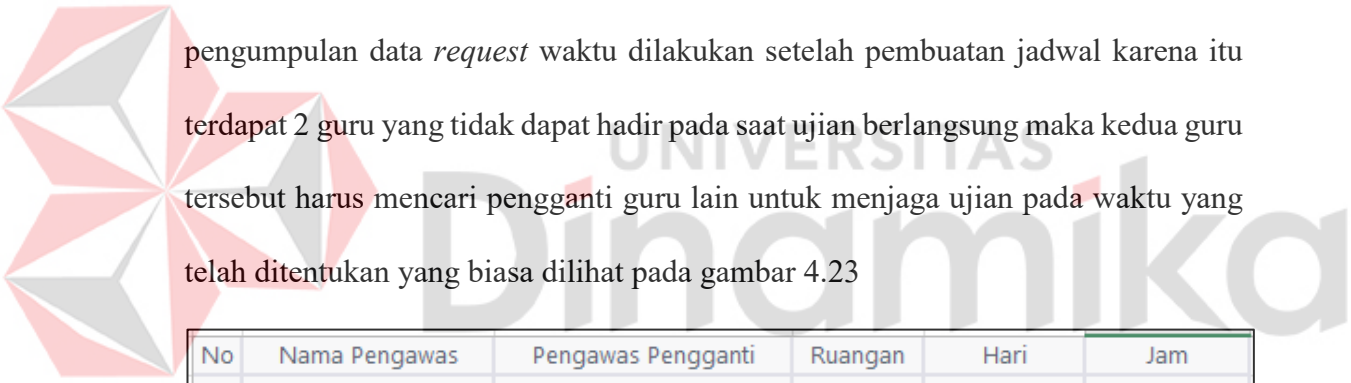
Gambar 4.22. Waktu Yang Dibutuhkan

3. Akurasi Penjadwalan

Pengujian akurasi pada pembuatan jadwal ujian manual dan sistem dengan mengacu pada tingkat keberhasilan atau ketepatan dalam menyusun jadwal ujian, namun saat dilakukan penjadwalan secara manual mengalami dua kali *crash*. Dalam konteks ini, "*crash*" mengindikasikan adanya kegagalan atau bentrokan

dalam menyusun jadwal ujian yang menyebabkan ketidaksesuaian atau tumpang tindih antara beberapa jadwal. Untuk menghitung akurasi, jumlah total jadwal ujian yang direncanakan secara manual dan jumlah *crash* yang terjadi harus diidentifikasi. Akurasi dihitung sebagai perbandingan antara jumlah jadwal yang berhasil direncanakan dengan jumlah total jadwal yang direncanakan secara manual dengan sistem, dan hasilnya dinyatakan sebagai persentase.

Dalam proses pengujian akurasi, penelitian ini akan membandingkan akurasi antara pembuatan jadwal ujian secara manual yang mengalami *crash* dua kali dalam 11 jadwal ujian, dengan sistem yang tidak mengalami *crash* sama sekali. Adanya *crash* jadwal pada jadwal ujian yang dibuat secara manual yang diakibatkan oleh pengumpulan data *request* waktu dilakukan setelah pembuatan jadwal karena itu terdapat 2 guru yang tidak dapat hadir pada saat ujian berlangsung maka kedua guru tersebut harus mencari pengganti guru lain untuk menjaga ujian pada waktu yang telah ditentukan yang biasa dilihat pada gambar 4.23



No	Nama Pengawas	Pengawas Pengganti	Ruangan	Hari	Jam
1	Eni Purwantini	Efi Triana Ningrum, S.Pd.	RB001	Jum'at	07:00 - 08:00
2	Prangripta Wibawa	Sunarmi, S.Pd.	RB002	Rabu	08:00 - 09:00

Gambar 4.23. *Request* Pengawas Pengganti

Akurasi Pembuatan Jadwal Ujian Manual dengan *Crash* Total jadwal ujian yang direncanakan secara manual 11 jadwal, Jumlah *crash* yang terjadi 2 jadwal.



Akurasi = (Total jadwal ujian - Jumlah *crash*) / Total jadwal ujian

Akurasi = (11 - 2) / 11

Akurasi = 9 / 11

Akurasi \approx 0.8181818181818182 (atau sekitar 82%)

Untuk akurasi sistem tanpa *crash* jadwal, total jadwal ujian yang direncanakan oleh sistem 11 jadwal, jumlah *crash* yang terjadi 0 jadwal yang dapat dilihat pada gambar 4.24. Eni Purwanti dan Prangripta Wibawa tidak menjadi pengawas pada hari rabu untuk Prangripta Wibawa dan pada hari jum'at untuk Eni Purwanti pada percobaan terakhir.

No	Jurusan	Ruang	Kelas	Mata Pelajaran	Waktu	Guru Pengampu	Aksi
1	TKJ	RB004	10	Bahasa Inggris	rabu, 09:15:00-10:15:00	Prangripta Wibawa	
2	ATPH	RB005	10	Alat Mesin Pertanian	rabu, 15:00:00-16:00:00	Prangripta Wibawa	

Gambar 4.24. Hasil Jadwal Ujian

Dan pada gambar 4.25. Eni Purwanti tidak menjadi pengawas pada hari jum'at untuk Eni Purwanti pada percobaan terakhir.

No	Jurusan	Ruang	Kelas	Mata Pelajaran	Waktu	Guru Pengampu	Aksi
1	KKLP	RB001	11	Pengelasan dan Pembubutan	selasa, 08:00:00-09:00:00	Eni Purwanti	
2	DKV	RB002	11	Matematika	senin, 10:15:00-11:15:00	Eni Purwanti	
3	DKV	RB002	12	Komputer Grafis	rabu, 12:30:00-13:30:00	Eni Purwanti	
4	TP	RB003	10	Simulasi dan Komunikasi Digital	rabu, 14:00:00-15:00:00	Eni Purwanti	
5	TKJ	RB004	10	Bahasa Indonesia	senin, 15:00:00-16:00:00	Eni Purwanti	

Gambar 4.25. Hasil Jadwal Ujian

Akurasi = (Total jadwal ujian - Jumlah *crash*) / Total jadwal ujian

Akurasi = (11 - 0) / 11

Akurasi = 11 / 11

Akurasi = 1 (atau 100%)

Akurasi pembuatan jadwal ujian secara manual yang mengalami *crash* adalah sekitar 82%. Akurasi sistem yang tidak mengalami *crash* adalah 100%. Dengan demikian, sistem yang tidak mengalami *crash* memiliki akurasi yang lebih tinggi dalam menyusun jadwal ujian dibandingkan dengan pembuatan jadwal secara manual yang mengalami *crash*.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah melakukan rancang bangun aplikasi penjadwalan sekolah dengan menggunakan algoritma genetika serta melakukan uji coba dapat disimpulkan sebagai berikut:

- 1) Sistem penjadwalan memiliki kelebihan dalam kecepatan penyusunan jadwal dan akurasi yang lebih tinggi dari pada proses manual dan dapat menghindari *crash* jadwal. Meski proses manual juga dapat berhasil menyusun jadwal dengan baik, sistem otomatis tidak hanya lebih akurat tetapi juga menghindari potensi kegagalan atau bentrokan dalam menyusun jadwal ujian. Oleh karena itu, penggunaan sistem penjadwalan ini memberikan dampak positif dalam penyusunan jadwal SMK 1 Trowulan yang akan menggunakannya. Hasil penelitian menunjukkan bahwa aplikasi penjadwalan pada SMK 1 Trowulan menggunakan algoritma genetika dapat mencapai tujuan dari penelitian ini.
- 2) Hasil penelitian menunjukkan bahwa aplikasi penjadwalan sekolah ini telah menghasilkan jadwal yang ditampilkan setiap satu minggu sesuai dengan syarat dan ketentuan yang berlaku.

5.2. Saran

Berikut ini merupakan saran untuk setelah melakukan rancang bangun aplikasi penjadwalan pada SMK 1 Trowulan.

- 1) Proses manual dapat mengumpulkan data permintaan jadwal guru sebelum melakukan proses penjadwalan.
- 2) Untuk mengoptimalkan fungsi *fitness* dapat ditambahkan algoritma *unsupervised learning* dalam proses fungsi *fitness* pada sistem penjadwalan. Beberapa algoritma *unsupervised learning* yaitu: *K-Means Clustering*, *Hierarchical Clustering* dan *Gaussian Mixture Model* (GMM).

DAFTAR PUSTAKA

- Farida, N. I. (2008). *Sistem Pendukung Keputusan Penjadwalan Pengajar Praktikum Laboratorium Komputer STIKOM Menggunakan Algoritma Genetika*. Surabaya: STIKOM Surabaya.
- Mahendra, R. S., Cobantoro, A. F., & Nurwanto. (2022). Rancang Bangun Aplikasi management Penjadwalan Perkuliahan Dengan Algoritma Genetika Berbasis Website. *Jurnal Rekayasa Teknologi dan Komputasi*.
- Pinedo, M. L. (2012). *Scheduling Theory, Algorithms, and Systems*. New York: Springer.
- Pressman, R. (2015). *Rekayasa Perangkat Lunak: Pendekatan Praktisi Buku 1*. Yogyakarta: Andi.
- Putranto, B. D., Utami, E., & Sunyoto, A. (2017). Perancangan Sistem Penjadwalan Ujian Menggunakan Algoritma Genetika Pada STIMIK Amikom Purwokerto. *Jurnal Telematika*.
- Ridwan, R., Kustian, N., & Ambarsa, E. W. (2022). Peran Data Store Dalam Mempresentasikan Hubungan Data Flow Diagram SSADM Dengan Entity Relationship Diagram. *Jurnal Ilmiah Teknik Mesin, Elektro Dan Komputer*.
- Sasongko, I. A. (2020). Rancang Bangun Aplikasi Penjadwalan Mata Pelajaran pada SMA GIKI 2 Surabaya. *Jsika*.
- Sutabri, T. (2012). *Analisis Sistem Informasi*. Yogyakarta: Andi.
- Suyatno. (2014). Artificial Intelligent. *Informatika Bandung*.
- Yakub. (2012). *Pengantar Sistem Informasi*. Yogyakarta: Graha Ilmu.