

## LAMPIRAN

### Lampiran 1. Code Proses Penjadwalan dengan Algoritma Genetika

app/Filament/Pages/ProsesPenjadwalan.php:

```
<?php
```

```
namespace App\Filament\Pages;
```

```
use App\Models\Pengampu;  
use App\Models\Ruang;  
use App\Models\Waktu;  
use App\Models\Jadwal;  
use Filament\Pages\Page;  
use Illuminate\Support\Facades\DB;  
use Illuminate\Support\Facades\Log;
```

```
class ProsesPenjadwalan extends Page
```

```
{  
    protected static ?string $navigationIcon = 'heroicon-o-cog-6-tooth';  
    protected static string $view = 'filament.pages.proses-penjadwalan';  
    protected static ?string $title = ""; // kosongkan judul besar  
    protected static ?string $navigationLabel = 'Proses Penjadwalan'; // label  
    menu di sidebar
```

```
    // Variabel yang akan ditampilkan di halaman  
    public $hasil = []; // hasil jadwal untuk ditampilkan  
    public $akurasi = 0; // nilai akurasi (%)  
    public $waktuProses = '-'; // waktu proses dalam detik  
    public $info = []; // informasi log / validasi
```

```
    // Parameter Algoritma Genetika  
    private $POPULASI = 30; // jumlah populasi awal  
    private $GENERASI = 50; // jumlah generasi  
    private $PROB_MUTASI = 0.08; // probabilitas mutasi (10%)
```

```
    // Jam istirahat → slot ini tidak boleh dipakai  
    private $ISTIRAHAT_START = '13:00:00';
```

```

private $ISTIRAHAT_END = '14:00:00';

public function mount()
{
    $this->info = session('jadwal_info', []);
    // Saat halaman dibuka, load data jadwal dari DB untuk ditampilkan
    $this->loadFromDatabase();
}

/**
 * Ambil data jadwal dari database dan siapkan untuk ditampilkan.
 */
private function loadFromDatabase()
{
    $query = Jadwal::query()
        ->join('waktus', 'jadwals.waktu_id', '=', 'waktus.id')
        ->select('jadwals.*')
        ->with([
            'pengampu.dosen',
            'pengampu.matakuliah',
            'pengampu.prodi',
            'ruangan',
            'waktu'
        ])
        ->orderByRaw("FIELD(waktus.hari,
'Senin','Selasa','Rabu','Kamis','Jumat','Sabtu')")
        ->orderBy('jadwals.jam_mulai');

    // mapping hasil supaya mudah ditampilkan di view
    $this->hasil = $query->get()->map(function ($j) {
        return [
            'hari'      => $j->waktu->hari ?? '-',
            // tampilkan jam yang disimpan di jadwals (hasil GA)
            'waktu'     => ($j->jam_mulai ?? $j->waktu->jam_mulai ?? '-') . '
s/d ' . ($j->jam_selesai ?? $j->waktu->jam_selesai ?? '-'),
            'matakuliah' => $j->pengampu->matakuliah->nama_matakuliah
            ?? '-',
            'sks'       => $j->pengampu->matakuliah->sks ?? '-',
            'semester'  => $j->pengampu->matakuliah->semester ?? '-',

```

```

        'grup'      => $j->pengampu->grup ?? '-',
        'dosen'    => $j->pengampu->dosen->nama_dosen ?? '-',
        'ruangan'  => $j->ruangan->nama_ruangan ?? '-',
        'program_studi' => $j->pengampu->prodi->nama_prodi ?? '-',
        'tahun_ajaran' => $j->tahun_ajaran ?? '-',
    ];
    })->toArray();
}

/**
 * Fungsi utama untuk menjalankan Algoritma Genetika, hitung evaluasi,
 * repair, simpan ke DB, dan tampilkan info.
 */
public function proses()
{
    // agar proses tidak timeout kalau dataset besar
    set_time_limit(1200);
    ini_set('memory_limit', '1536M');
    $start = microtime(true);

    try {
        // Ambil data dari database
        $pengampus = Pengampu::with(['dosen', 'matakuliah', 'prodi']->get());
        $ruangans = Ruangan::all();
        $waktusRaw = Waktu::orderBy('hari')->orderBy('jam_mulai')->get();

        // Validasi awal: data harus ada
        if ($pengampus->isEmpty() || $ruangans->isEmpty() || $waktusRaw->isEmpty()) {
            $this->info[] = "✘ Data pengampu, ruangan, atau waktu masih kosong.";
            session(['jadwal_info' => $this->info]);
            return;
        }

        // filter waktu valid (jam_mulai < jam_selesai)
        $waktus = $waktusRaw->filter(fn($w) => strtotime($w->jam_mulai) < strtotime($w->jam_selesai))->values();
        if ($waktus->isEmpty()) {

```

```

    $this->info[] = " ✘ Semua slot waktu tidak valid.";
    session(['jadwal_info' => $this->info]);
    return;
}

// Buat mapping waktu_id → index array waktus
$slotIndex = [];
foreach ($waktus as $idx => $slot) {
    $slotIndex[$slot->id] = $idx;
}

// Bentuk unit kuliah (setiap pengampu jadi 1 unit yang harus
dijadwalkan)
$units = [];
foreach ($pengampus as $p) {
    $units[] = [
        'unit_id'      => $p->id,
        'matakuliah'  => $p->matakuliah->nama_matakuliah ?? '-',
        'sks'          => (int)($p->matakuliah->sks ?? 2),
        'semester'    => $p->matakuliah->semester ?? '-',
        'dosen_ids'   => [$p->dosen_id],
        'dosen_nama'  => optional($p->dosen)->nama_dosen ?? '-',
        'groups'      => [$p->prodi->nama_prodi . '-G' . $p->grup],
        'grup'        => $p->grup ?? '-',
        'program_studi' => optional($p->prodi)->nama_prodi ?? '-',
        'tahun_ajaran' => $p->tahun_ajaran ?? '-',
        'jenis_ruangan' => $p->matakuliah->jenis_ruangan ?? 'Ruang
Kelas',
    ];
}

// Filter slot waktu valid (tidak melanggar jam istirahat)
$validWaktus = $waktus->filter(function ($w) {
    $s = strtotime($w->jam_mulai);
    $e = strtotime($w->jam_selesai);
    $istS = strtotime($this->ISTIRAHAT_START);
    $istE = strtotime($this->ISTIRAHAT_END);
    return !($s < $istE && $e > $istS);
})->values();

```

```

if ($validWaktus->isEmpty()) {
    $this->info[] = "✘ Semua slot waktu bentrok jam istirahat.";
    session(['jadwal_info' => $this->info]);
    return;
}

// === 1. Pembentukan Populasi Awal ===
$populasi = [];
for ($i = 0; $i < $this->POPULASI; $i++) {
    $populasi[] = $this->randomChromosome($units, $ruangans,
$validWaktus);
}

$bestChrom = null;
$bestEval = null;

// === 2. Evolusi Generasi ===
for ($g = 0; $g < $this->GENERASI; $g++) {
    $evaluasi = [];
    foreach ($populasi as $chrom) {
        // evaluasi fitness setiap kromosom
        $evaluasi[] = $this->evaluateChromosome($chrom, $units,
$ruangans, $waktus, $slotIndex);
    }

    // Simpan log di storage/logs/laravel.log
    $avgFitness = round(collect($evaluasi)->avg('score'), 2);
    $maxFitness = round(collect($evaluasi)->max('score'), 2);
    $totalHard = collect($evaluasi)->sum('hard_violations');
    Log::info("Generasi " . ($g + 1) . " | Rata2 Fitness: {$avgFitness} |
Maks: {$maxFitness} | Total Hard Violations: {$totalHard}");

    // Atau tampilkan ke browser kalau dijalankan via Filament
    echo "Generasi " . ($g + 1) . ": Fitness Rata-rata = {$avgFitness},
Maks = {$maxFitness}, Pelanggaran = {$totalHard}<br>";

    // simpan yang terbaik
    foreach ($evaluasi as $i => $e) {

```

```

        if (!$bestEval || $e['score'] > $bestEval['score']) {
            $bestEval = $e;
            $bestChrom = $populasi[$i];
        }
    }

    // === 3. Seleksi + Crossover + Mutasi ===
    $populasi = $this->nextGeneration($populasi, $evaluasi, $units,
    $ruangans, $validWaktus);
}

if (!$bestChrom) {
    $this->info[] = "✘ Tidak ada solusi valid.";
    session(['jadwal_info' => $this->info]);
    return;
}

// === 4. Repair Solusi Terbaik ===
$bestChrom = $this->repairChromosome($bestChrom, $units,
$ruangans, $waktus)['chrom'];

// evaluasi terakhir
$finalEval = $this->evaluateChromosome($bestChrom, $units,
$ruangans, $waktus, $slotIndex);

// === 5. Tampilkan hasil ke view ===
$this->hasil = $this->expandAssignmentsForDisplay($bestChrom,
$units, $ruangans, $waktus);

// === 6. Hitung akurasi ===
$this->akurasi = round(($finalEval['constraints_satisfied'] / max(1,
$finalEval['total_constraints'])) * 100, 2);
$this->waktuProses = round(microtime(true) - $start, 3) . ' detik';

// info log
$this->info[] = "☑ Jadwal selesai.";
$this->info[] = "Hard violations: {$finalEval['hard_violations']}";
$this->info[] = "Constraint terpenuhi:
{$finalEval['constraints_satisfied']} dari {$finalEval['total_constraints']}";

```

```

$this->info[] = "Akurasi: {$this->akurasi}% | Waktu proses: {$this->waktuProses}";

// simpan log ke session
session([
    'jadwal_info' => $this->info,
    'jadwal_akurasi' => $this->akurasi,
    'jadwal_waktu_proses' => $this->waktuProses,
]);

// === 7. Simpan ke Database ===
DB::transaction(function () use ($bestChrom, $units, $waktus,
    $slotIndex, $ruangans) {
    // gunakan delete() bukan truncate() untuk menghindari issue
    transaction/foreign key
    Jadwal::query()->delete(); // kosongkan jadwal lama

    foreach ($bestChrom as $row) {
        $unit = collect($units)->firstWhere('unit_id', $row['unit_id']);
        $waktuAwal = $waktus->firstWhere('id', $row['waktu_id']);
        $ruangan = $ruangans->firstWhere('id', $row['ruangan_id']);

        // safety: jika salah satu tidak valid, pilih ulang dari
        validWaktus/ruangans
        if (!$unit || !$ruangan || !$waktuAwal) {
            // skip penyimpanan unit yang invalid (atau pilih fallback)
            continue;
        }

        // tentukan jam selesai berdasarkan SKS (cari consecutive slots)
        $sks = max(1, (int)$unit['sks']);
        $startIndex = $slotIndex[$waktuAwal->id] ?? $waktus->search(fn($x) => $x->id == $waktuAwal->id);
        $endSlot = $waktuAwal;
        for ($i = 1; $i < $sks; $i++) {
            $next = $waktus->get($startIndex + $i);
            if (!$next || $next->hari !== $waktuAwal->hari) break;
            $prev = $waktus->get($startIndex + $i - 1);
            if (!$prev || strtotime($prev->jam_selesai) !== strtotime($next->

```

```

>jam_mulai)) break;
        // jangan lewat jam istirahat
        if (strtotime($next->jam_mulai) < strtotime($this-
>ISTIRAHAT_END) && strtotime($next->jam_selesai) > strtotime($this-
>ISTIRAHAT_START)) break;
        $endSlot = $next;
    }

    // simpan ke DB
    Jadwal::create([
        'pengampu_id' => $row['unit_id'],
        'ruangan_id' => $row['ruangan_id'],
        'waktu_id' => $row['waktu_id'],
        'jam_mulai' => $waktuAwal->jam_mulai,
        'jam_selesai' => $endSlot->jam_selesai,
        'tahun_ajaran' => $unit['tahun_ajaran'] ?? '-',
    ]);
    }
});

$this->loadFromDatabase(); // reload untuk sinkron

} catch (\Throwable $e) {
    // tampilkan error agar bisa debug. Jika terjadi di DB::transaction,
    Laravel akan rollback otomatis.
    $this->info[] = "ERROR: " . $e->getMessage();
    session(['jadwal_info' => $this->info]);
}
}

// ===== HELPER FUNCTIONS =====

/**
 * Buat kromosom random (populasi awal)
 * setiap unit dipasangkan ruangan (yang sesuai jenis) dan waktu
(validWaktus)
 */
private function randomChromosome($units, $ruangans, $validWaktus):
array

```

```

{
    $chrom = [];
    foreach ($units as $u) {
        // pilih ruangan sesuai jenis
        $ruanganList = $ruangans->where('jenis_ruangan',
$u['jenis_ruangan']);
        $ruangan = $ruanganList->isNotEmpty() ? $ruanganList->random() :
$ruangans->random();
        // pilih waktu random (tidak bentrok istirahat)
        $waktu = $validWaktus->random();
        $chrom[] = [
            'unit_id' => $u['unit_id'],
            'ruangan_id' => $ruangan->id,
            'waktu_id' => $waktu->id
        ];
    }
    return $chrom;
}

/**
 * Evaluasi kromosom → hitung fitness
 * Constraint yang dicek:
 * - SKS harus consecutive slot
 * - Tidak bentrok ruangan
 * - Tidak bentrok dosen
 * - Tidak bentrok grup
 */
private function evaluateChromosome($chrom, $units, $ruangans,
$waktus, $slotIndex = []): array
{
    $shard = 0; // jumlah hard violation
    $satisfied = 0; // constraint terpenuhi
    $total = 0; // total constraint

    // tabel occupancy
    $occupiedRoom = [];
    $occupiedDosen = [];
    $occupiedGroup = [];

```

```

foreach ($chrom as $assign) {
    $unit = collect($units)->firstWhere('unit_id', $assign['unit_id']);
    $r = $ruangans->firstWhere('id', $assign['ruangan_id']);
    $w = $waktus->firstWhere('id', $assign['waktu_id']);

    if (!$unit || !$r || !$w) {
        // Kemungkinan unit/ruangan/waktu invalid -> penalti besar
        $hard += 3;
        continue;
    }

    $sks = max(1, (int)$unit['sks']);
    $startIndex = $slotIndex[$w->id] ?? $waktus->search(fn($x) => $x->id
== $w->id);
    if ($startIndex === false) {
        $hard += 3;
        continue;
    }

    // cek per slot sesuai SKS
    for ($i = 0; $i < $sks; $i++) {
        $total++; // tambah 1 constraint
        $slot = $waktus->get($startIndex + $i);
        if (!$slot || $slot->hari != $w->hari) {
            $hard++;
            continue;
        }
        if ($i > 0) {
            $prev = $waktus->get($startIndex + $i - 1);
            if (!$prev || strtotime($prev->jam_selesai) != strtotime($slot-
>jam_mulai)) {
                $hard++;
                continue;
            }
        }
    }

    // jam istirahat check
    $istS = strtotime($this->ISTIRAHAT_START);
    $istE = strtotime($this->ISTIRAHAT_END);

```

```

        if (strtotime($slot->jam_mulai) < $istE && strtotime($slot->
jam_selesai) > $istS) {
            $hard++;
            continue;
        }

        // cek bentrok ruangan/dosen/grup
        $rw = $r->id . '-' . ($startIndex + $i);
        $dw = $unit['dosen_ids'][0] . '-' . ($startIndex + $i);
        $gw = $unit['groups'][0] . '-' . ($startIndex + $i);
        if (isset($occupiedRoom[$rw]) || isset($occupiedDosen[$dw]) ||
isset($occupiedGroup[$gw])) {
            $hard++;
            continue;
        }
        $occupiedRoom[$rw] = true;
        $occupiedDosen[$dw] = true;
        $occupiedGroup[$gw] = true;
        $satisfied++;
    }
}

// fitness score = constraint terpenuhi - penalti * 10
$score = $satisfied - $hard * 10;
return [
    'score' => $score,
    'hard_violations' => $hard,
    'constraints_satisfied' => $satisfied,
    'total_constraints' => $total
];
}

/**
 * Membuat generasi baru → seleksi + crossover + mutasi
 * Implementasi sederhana: elitism + one-point-like crossover + random
mutation.
 */
private function nextGeneration($pop, $eval, $units, $ruangans,
$validWaktus): array

```

```

{
    // urutkan berdasarkan fitness (score)
    array_multisort(array_column($eval, 'score'), SORT_DESC, $pop,
$eval);

    // elitism: ambil 40-60% terbaik
    $eliteCount = max(1, (int) ceil(count($pop) * 0.5));
    $elite = array_slice($pop, 0, $eliteCount);

    $new = $elite;

    // isi populasi dengan crossover + mutasi
    while (count($new) < count($pop)) {
        $p1 = $elite[array_rand($elite)];
        $p2 = $elite[array_rand($elite)];
        $child = [];

        foreach ($p1 as $i => $gene) {
            // crossover 1 point sederhana
            // ambil gen dari p1 atau p2
            $child[$i] = rand(0, 1) ? $gene : ($p2[$i] ?? $gene);
        }
        // mutasi → ubah slot ruangan/waktu secara acak
        if (mt_rand() / mt_getrandmax() < $this->PROB_MUTASI) {
            $idx = array_rand($child);
            $child[$idx]['ruangan_id'] = $ruangans->random()->id;
            $child[$idx]['waktu_id'] = $validWaktus->random()->id;
        }
        $new[] = $child;
    }
    return $new;
}

/**
 * Repair kromosom pintar:
 * - Jika ada bentrok/invalid, cari slot waktu & ruangan baru yang valid
 * - Valid = tidak bentrok ruangan/dosen/grup, sesuai SKS, dan tidak
melewati jam istirahat
 */

```

```

private function repairChromosome($chrom, $units, $ruangans, $waktus):
array
{
    $occupiedRoom = [];
    $occupiedDosen = [];
    $occupiedGroup = [];

    foreach ($chrom as &$assign) {
        $unit = collect($units)->firstWhere('unit_id', $assign['unit_id']);
        if (!$unit) continue;

        $valid = false;

        foreach ($ruangans as $r) {
            // ruangan harus sesuai jenis
            if ($r->jenis_ruangan !== $unit['jenis_ruangan']) continue;

            foreach ($waktus as $i => $w) {
                $sks = max(1, (int)$unit['sks']);
                $endSlot = $w;
                $ok = true;

                // cek SKS consecutive + tidak kena istirahat
                for ($k = 1; $k < $sks; $k++) {
                    $next = $waktus->get($i + $k);
                    if (!$next || $next->hari !== $w->hari) {
                        $ok = false;
                        break;
                    }
                    $prev = $waktus->get($i + $k - 1);
                    if (strtotime($prev->jam_selesai) !== strtotime($next->jam_mulai)) {
                        $ok = false;
                        break;
                    }
                }
                // cek jam istirahat
                if (
                    strtotime($next->jam_mulai) < strtotime($this->ISTIRAHAT_END) &&

```

```

        strtotime($next->jam_selesai) > strtotime($this-
>ISTIRAHAT_START)
    ) {
        $ok = false;
        break;
    }
    $endSlot = $next;
}

if (!$ok) continue;

// cek bentrok dengan jadwal yang sudah ditempatkan
$conflict = false;
for ($k = 0; $k < $sks; $k++) {
    $slot = $waktus->get($i + $k);
    if (!$slot) {
        $conflict = true;
        break;
    }

    $rw = $r->id . '-' . ($i + $k);
    $dw = $unit['dosen_ids'][0] . '-' . ($i + $k);
    $gw = $unit['groups'][0] . '-' . ($i + $k);

    if (isset($occupiedRoom[$rw]) || isset($occupiedDosen[$dw]) ||
isset($occupiedGroup[$gw])) {
        $conflict = true;
        break;
    }
}

if ($conflict) continue;

//  jika semua valid → pakai slot ini
$assign['ruangan_id'] = $r->id;
$assign['waktu_id'] = $w->id;

// tandai occupancy
for ($k = 0; $k < $sks; $k++) {

```

```

        $slot = $waktus->get($i + $k);
        if ($slot) {
            $occupiedRoom[$r->id . '-' . ($i + $k)] = true;
            $occupiedDosen[$unit['dosen_ids'][0] . '-' . ($i + $k)] = true;
            $occupiedGroup[$unit['groups'][0] . '-' . ($i + $k)] = true;
        }
    }

    $valid = true;
    break 2; // keluar dari loop waktu & ruangan
}

if (!$valid) {
    // fallback → random supaya tidak kosong
    $ruangan = $ruangans->random();
    $waktu = $waktus->random();
    $assign['ruangan_id'] = $ruangan->id;
    $assign['waktu_id'] = $waktu->id;
}

return ['chrom' => $chrom];
}

/**
 * Ekspansi jadwal untuk ditampilkan (gabungkan slot sesuai SKS)
 */
private function expandAssignmentsForDisplay($chrom, $units, $ruangans,
$waktus)
{
    $hasil = [];
    foreach ($chrom as $assign) {
        $unit = collect($units)->firstWhere('unit_id', $assign['unit_id']);
        $ruangan = $ruangans->firstWhere('id', $assign['ruangan_id']);
        $waktu = $waktus->firstWhere('id', $assign['waktu_id']);
        if (!$unit || !$ruangan || !$waktu) continue;

        $sks = max(1, (int)$unit['sks']);

```

```

$startIndex = $waktus->search(fn($x) => $x->id == $waktu->id);
$endSlot = $waktu;

// gabungkan beberapa slot sesuai jumlah SKS
if ($startIndex !== false) {
    for ($i = 1; $i < $sks; $i++) {
        $next = $waktus->get($startIndex + $i);
        if (!$next || $next->hari !== $waktu->hari) break;
        $prev = $waktus->get($startIndex + $i - 1);
        if (!$prev || strtotime($prev->jam_selesai) !== strtotime($next->
jam_mulai)) break;
        // hindari melewati istirahat
        if (strtotime($next->jam_mulai) < strtotime($this->
ISTIRAHAT_END) && strtotime($next->jam_selesai) > strtotime($this->
ISTIRAHAT_START)) break;
        $endSlot = $next;
    }
}

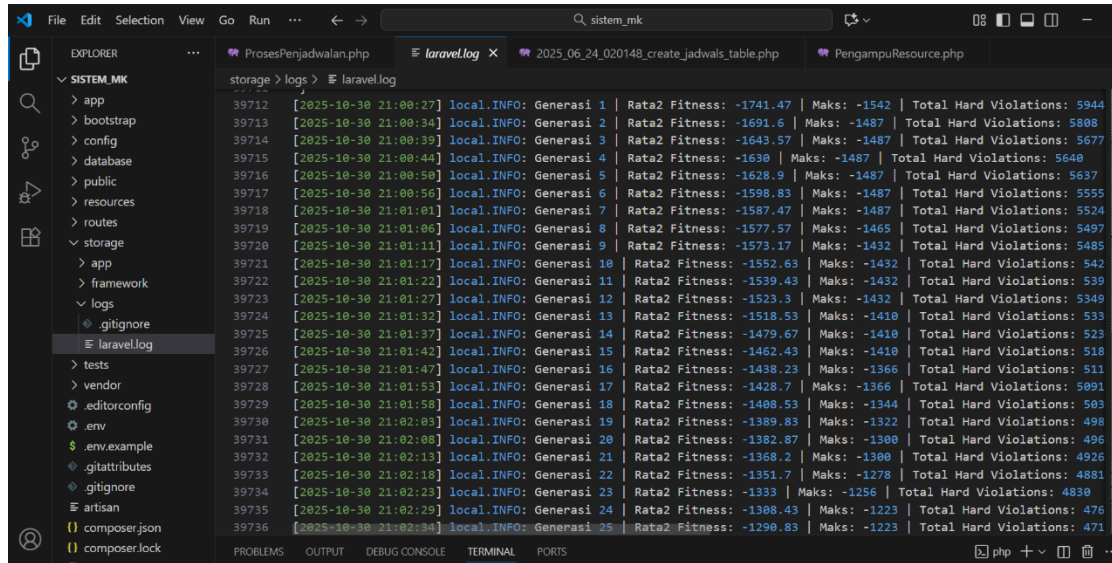
$mulai = $waktu->jam_mulai ?? '-';
$selesai = $endSlot->jam_selesai ?? '-';

// masukkan ke hasil untuk ditampilkan di tabel Filament
$hasil[] = [
    'hari' => $waktu->hari ?? '-',
    'waktu' => ($mulai ?? '-') . ' s/d ' . ($selesai ?? '-'),
    'matakuliah' => $unit['matakuliah'],
    'sks' => $sks,
    'semester' => $unit['semester'] ?? '-',
    'grup' => $unit['grup'] ?? '-',
    'dosen' => $unit['dosen_nama'] ?? '-',
    'ruangan' => $ruangan->nama_ruangan ?? '-',
    'program_studi' => $unit['program_studi'] ?? '-',
    'tahun_ajaran' => $unit['tahun_ajaran'] ?? '-',
];
}
return $hasil;
}

```

}

## Lampiran 2 Log Iterasi Algoritma Genetika



```
storage > logs > laravel.log
39712 [2025-10-30 21:00:27] local.INFO: Generasi 1 | Rata2 Fitness: -1741.47 | Maks: -1542 | Total Hard Violations: 5944
39713 [2025-10-30 21:00:34] local.INFO: Generasi 2 | Rata2 Fitness: -1691.6 | Maks: -1487 | Total Hard Violations: 5808
39714 [2025-10-30 21:00:39] local.INFO: Generasi 3 | Rata2 Fitness: -1643.57 | Maks: -1487 | Total Hard Violations: 5677
39715 [2025-10-30 21:00:44] local.INFO: Generasi 4 | Rata2 Fitness: -1630 | Maks: -1487 | Total Hard Violations: 5640
39716 [2025-10-30 21:00:50] local.INFO: Generasi 5 | Rata2 Fitness: -1628.9 | Maks: -1487 | Total Hard Violations: 5637
39717 [2025-10-30 21:00:56] local.INFO: Generasi 6 | Rata2 Fitness: -1598.83 | Maks: -1487 | Total Hard Violations: 5555
39718 [2025-10-30 21:01:01] local.INFO: Generasi 7 | Rata2 Fitness: -1587.47 | Maks: -1487 | Total Hard Violations: 5524
39719 [2025-10-30 21:01:06] local.INFO: Generasi 8 | Rata2 Fitness: -1577.57 | Maks: -1465 | Total Hard Violations: 5497
39720 [2025-10-30 21:01:11] local.INFO: Generasi 9 | Rata2 Fitness: -1573.17 | Maks: -1432 | Total Hard Violations: 5485
39721 [2025-10-30 21:01:17] local.INFO: Generasi 10 | Rata2 Fitness: -1552.63 | Maks: -1432 | Total Hard Violations: 542
39722 [2025-10-30 21:01:22] local.INFO: Generasi 11 | Rata2 Fitness: -1539.43 | Maks: -1432 | Total Hard Violations: 539
39723 [2025-10-30 21:01:27] local.INFO: Generasi 12 | Rata2 Fitness: -1523.3 | Maks: -1432 | Total Hard Violations: 5349
39724 [2025-10-30 21:01:32] local.INFO: Generasi 13 | Rata2 Fitness: -1518.53 | Maks: -1410 | Total Hard Violations: 533
39725 [2025-10-30 21:01:37] local.INFO: Generasi 14 | Rata2 Fitness: -1479.67 | Maks: -1410 | Total Hard Violations: 523
39726 [2025-10-30 21:01:42] local.INFO: Generasi 15 | Rata2 Fitness: -1462.43 | Maks: -1410 | Total Hard Violations: 518
39727 [2025-10-30 21:01:47] local.INFO: Generasi 16 | Rata2 Fitness: -1438.23 | Maks: -1366 | Total Hard Violations: 511
39728 [2025-10-30 21:01:53] local.INFO: Generasi 17 | Rata2 Fitness: -1428.7 | Maks: -1366 | Total Hard Violations: 5091
39729 [2025-10-30 21:01:58] local.INFO: Generasi 18 | Rata2 Fitness: -1408.53 | Maks: -1344 | Total Hard Violations: 503
39730 [2025-10-30 21:02:03] local.INFO: Generasi 19 | Rata2 Fitness: -1389.83 | Maks: -1322 | Total Hard Violations: 498
39731 [2025-10-30 21:02:08] local.INFO: Generasi 20 | Rata2 Fitness: -1382.87 | Maks: -1300 | Total Hard Violations: 496
39732 [2025-10-30 21:02:13] local.INFO: Generasi 21 | Rata2 Fitness: -1368.2 | Maks: -1300 | Total Hard Violations: 4926
39733 [2025-10-30 21:02:18] local.INFO: Generasi 22 | Rata2 Fitness: -1351.7 | Maks: -1278 | Total Hard Violations: 4881
39734 [2025-10-30 21:02:23] local.INFO: Generasi 23 | Rata2 Fitness: -1333 | Maks: -1256 | Total Hard Violations: 4830
39735 [2025-10-30 21:02:29] local.INFO: Generasi 24 | Rata2 Fitness: -1308.43 | Maks: -1223 | Total Hard Violations: 476
39736 [2025-10-30 21:02:34] local.INFO: Generasi 25 | Rata2 Fitness: -1290.83 | Maks: -1223 | Total Hard Violations: 471
```

## Lampiran 3 Data Uji

### a. Data Daftar Dosen

DAFTAR DOSEN PENGAMPU MATA KULIAH :			
<b>BIN</b>	Binsar Sihombing,SE,M.Si	<b>APH</b>	Aji Prasetya Hadi,S.Par,MM.Par
<b>FLS</b>	Ferry Laurensius,SE,M.Si	<b>RIZ</b>	Rizky Ramadhansyah,ST.,M.Kom
<b>VIN</b>	Drs. Vinsensius Matondang,MBA	<b>DAV</b>	David Gordon
<b>JEN</b>	Dr. Jenny Elisabeth	<b>SEP</b>	Septian Simatupang,S.Kom,M.Kom
<b>ALD</b>	Dr. Aldon Marungkap H.P. Sinaga	<b>TFQ</b>	Taufiqurrahman, ST., M.Kom
<b>AST</b>	Aston L. Situmorang,SE,M.Si	<b>NAS</b>	Nora Anisa Br. Sinulingga,SE,MM
<b>HAM</b>	Hendra Agustinus H. Marbun,SE,M.Si	<b>EEH</b>	Enty Evasari Hot Asi Simanjuntak,SE,M.S.M.
<b>BIL</b>	Bilson Pandiangan,S.Si,MM	<b>ROE</b>	Muhammad Roestian Fahmi Nasution,S.Ik.,MM
<b>AME</b>	Amelira Haris Nasution,SP,M.Si	<b>RFT</b>	Rio Fernandez Tamba,S.Pd, M.Sn.
<b>AWP</b>	Asmara Wildani Pasaribu,S.Sos,MM	<b>BES</b>	Bernadet Sihombing,S.M.,M.Sc.
<b>FEZ</b>	Ferawati,SE,M.Si,CA	<b>FRE</b>	Dr. Frederik Situmeang
<b>NAT</b>	Natalia Lusianingsih Sumanto, S.Si, M.Si	<b>BEN</b>	Dr. Benny Bernardus
<b>ALB</b>	Albert Sembiring, S.Si, M.Si	<b>PUR</b>	Juliansen Purba
<b>NIL</b>	Nilam Kemal Odang,SE,M.Acc.,Ak.,CPA	<b>BAH</b>	Syaiful Bahri Panjaitan, SP., M. Agric Sc
<b>TGH</b>	Dr. Teguh Triono	<b>NIR</b>	Nirmala Purba,SE,MA
<b>DAJ</b>	Dr. Dany Juhandi	<b>AGB</b>	Drs. Argensia Bangun,M.Si
<b>NNA</b>	Nukeu Novia Andriani S.,S.S,MM.Par	<b>ANT</b>	Anthon Sitepu
<b>MMT</b>	Dr. Mukdin Markus Turnip	<b>JER</b>	Jeremi Tarigan ,S.Kom
<b>MAR</b>	Maryam Monica Rangkuti,SE,M.Si	<b>PHS</b>	Dr. Patar Hasudongan Silitonga
<b>MUS</b>	Muslim Ramli,S.Kom,M.Kom	<b>AZS</b>	Alvin Zonatan Sagala,S.Pd.,M.Li
<b>ICS</b>	Indah Clara Sari,S.Kom,M.Kom	<b>KOS</b>	Kosman Samosir,SH,M.Hum
<b>JAG</b>	Juni Anggraini,SST,M.Si		

### b. Data Jadwal Genap 2024/2025

The screenshot displays a web browser window with the following details:

- Browser:** Google Chrome
- Address Bar:** File: D:/0.%20PROPOSAL%20TA/1.%20FILE%20TUGAS%20AKHIR/JADWAL%20GENAP%202024%202025%20REV%201.pdf
- Page Title:** JADWAL PERKULIAHAN SEMESTER GENAP T.A 2024 / 2025
- Table Content:** A large table with multiple columns, including course codes (e.g., IKT101, IKT102), course titles (e.g., Pengantar Sistem Informasi, Sistem Basis Data), lecturers (e.g., Binsar Sihombing, Ferry Laurensius), and schedule details (e.g., days and times).

c. Data Ruangan

No	Ruangan	Lokasi	Kapasitas
1	Kelas 3.1	Lantai 3	0
2	Lab. Komputer	Lantai 3	30
3	Kelas 3.2	Lantai 3	36
4	Kelas 3.3	Lantai 3	36
5	Mini teater	Lantai 3	96
6	Kelas 3.4	Lantai 3	36
7	Kelas 3.5	Lantai 3	40
8	Kelas 3.6	Lantai 3	40
9	Kelas 3.7	Lantai 3	40
10	Kelas 3.8	Lantai 3	64
11	Kelas 3.9	Lantai 3	36
12	Kelas 3.10	Lantai 3	36
13	Lab. Akuntansi	Lantai 3	15
14	Kelas 3.11	Lantai 3	48
15	Kelas 4.1	Lantai 4	20
16	kelas 4.2	Lantai 4	48
17	Lab. Komputer	Lantai 6	30
18	Kelas 4.3	Lantai 4	70
19	Pelatihan	Lantai 4	48
20	Kelas 4.4	Lantai 4	40
21	Kelas 4.5	Lantai 4	170
22	Kelas 4.6	Lantai 4	30
23	Kelas 4.7	Lantai 4	30
24	Kelas 4.8	Lantai 4	47
25	Kelas 4.9	Lantai 4	30

## Lampiran 4. Hasil Pengujian di Sistem

The screenshot shows a web application interface for 'Politeknik WBI'. The browser address bar shows '127.0.0.1:8000/admin/proses-penjadwalan'. The page title is 'Politeknik WBI'. The main content area is titled 'Proses Penjadwalan' and features a green 'Proses Penjadwalan' button and an orange 'Refresh' button. Below this, there is an 'Info Validasi' section with the following details:

- ✓ Jadwal selesai.
- Hard violations: 90
- Constraint terpenuhi: 348 dari 438
- Akurasi: 79.45% | Waktu proses: 290.842 detik

Summary: **Akurasi: 79.45% | Waktu Proses: 290.842 detik**

Hari	Waktu	Mata Kuliah	SKS	Semester	Grup	Dosen	Ruangan	Program Studi	Tahun Ajaran
Senin	08:00:00 s/d 10:00:00	Manajemen	2	2	32	(PHSI) - Dr. Patar Hasudongon Silitorga	Ruang Kelas 3.5	Agribisnis Hortikultura	2024/20
Senin	08:00:00 s/d 10:00:00	Sikap Religius	2	4	24	(VINI) - Dra. Vinsensius Matondang, MBA	Ruang Kelas 3.1	Akuntansi Perpajakan	2024/20

## Lampiran 5. Hasil Ekspor PDF & Excel

### a. Hasil Ekspor PDF

The screenshot shows a PDF document titled "JADWAL PERKULIAHAN SEMESTER GENAP T.A 2024/2025 POLITEKNIK WBI". The document contains a table with the following columns: Hari, Jam, Mata Kuliah, SKS, Semester, Grup, Ruang, Dosen, Program Studi, and Tahun Ajaran. The table lists 20 courses for the second semester, including subjects like "Pengantar Model Bisnis", "Manajemen", "Perencanaan Jasa", "Pengembangan Model Bisnis", "Pemasaran", "Bahasa Indonesia", "Teori Ekonomi", "Matematika Keuangan dan Bisnis", "Akuntansi Usaha Dagang", "PPH dan Potput PPH", "Seni dan Olahraga", "Akuntansi Usaha Dagang", "Perencanaan Jasa", "Pengembangan Model Bisnis", "Manajemen", "Bahasa Indonesia", "Teori Ekonomi", "Matematika Keuangan dan Bisnis", "Akuntansi Usaha Dagang", "PPH dan Potput PPH", "Seni dan Olahraga", and "Akuntansi Usaha Dagang".

### b. Hasil Ekspor Excel

The screenshot shows an Excel spreadsheet titled "JADWAL PERKULIAHAN SEMESTER GENAP T.A 2024/2025 POLITEKNIK WBI". The spreadsheet contains a table with the following columns: Hari, Jam, Mata Kuliah, SKS, Semester, Grup, Ruang, Dosen, Prodi, and Tahun Ajaran. The table lists 20 courses for the second semester, including subjects like "Pengantar Model Bisnis", "Manajemen", "Perencanaan Jasa", "Pengembangan Model Bisnis", "Pemasaran", "Bahasa Indonesia", "Teori Ekonomi", "Matematika Keuangan dan Bisnis", "Akuntansi Usaha Dagang", "PPH dan Potput PPH", "Seni dan Olahraga", "Akuntansi Usaha Dagang", "Perencanaan Jasa", "Pengembangan Model Bisnis", "Manajemen", "Bahasa Indonesia", "Teori Ekonomi", "Matematika Keuangan dan Bisnis", "Akuntansi Usaha Dagang", "PPH dan Potput PPH", "Seni dan Olahraga", and "Akuntansi Usaha Dagang".