

APLIKASI PENJADWALAN PEMBELAJARAN SEKOLAH MENGUNAKAN MODEL PROTOTYPE (STUDI KASUS MI AL-MA'ARIF MARGOMULYO 02)

Sofi Nasiroh, Sri Lestanti, Saiful Nur Budiman

Program Studi Teknik Informatika, Universitas Islam Balitar

sofinasiroh21@gmail.com, lestanti85@gmail.com, sync.saifulnb@gmail.com

ABSTRAK

Penjadwalan pembelajaran di MI Al-Ma'arif Margomulyo 02 saat ini masih dilakukan secara manual, menyebabkan berbagai kendala seperti bentrok jadwal guru, kekosongan kelas, serta kesalahan dalam penentuan waktu dan ruang kelas. Penelitian ini bertujuan untuk merancang sebuah aplikasi penjadwalan pembelajaran sekolah berbasis web yang dapat memproses penjadwalan secara terkomputerisasi. Dengan penjadwalan yang terkomputerisasi, diharapkan admin sekolah dapat lebih mudah dalam melakukan pembagian jadwal. Metode penelitian yang digunakan adalah *Research And Development (RND) Model Prototype* yang mempermudah pengembang melakukan komunikasi dengan pengguna. Untuk menghasilkan jadwal yang optimal menggunakan algoritma optimasi yaitu algoritma genetika. Pengujian aplikasi dilakukan menggunakan tiga jenis pengujian, yaitu *blackbox testing*, *whitebox testing*, dan *beta testing*. Hasil penelitian menunjukkan bahwa aplikasi penjadwalan berhasil menggenerate jadwal pembelajaran dengan tingkat kesesuaian sebesar 97% berdasarkan pengujian *blackbox*. Pengujian *whitebox* termasuk *a simple procedure* dengan *resk* yaitu *low* dan hasil pengujian *beta testing* menunjukkan bahwa pengguna aplikasi memberikan tingkat kepuasan sebesar 80,1%.

Kata kunci: Aplikasi Penjadwalan, Prototype, Algoritma Genetika, Pengujian Aplikasi

1. PENDAHULUAN

Saat ini teknologi berkembang pesat, dan sistem informasi beroperasi sesuai dengan tingkat kebutuhan penggunaannya. Tidak terkecuali pada bidang teknologi informasi yang kini menjadi bagian tak terpisahkan dari kehidupan masyarakat. Hal ini disebabkan oleh banyaknya manfaat yang diperoleh dari perkembangan teknologi, salah satunya pada bidang pendidikan. Sekolah adalah salah satu contoh lembaga pendidikan yang menjadi tempat terjadinya kegiatan belajar mengajar dan kegiatan lain yang menunjang terselenggaranya proses belajar mengajar. Pengaturan jadwal pelajaran yang baik merupakan salah satu hal yang berperan penting dalam proses mengajar. Seorang guru yang mengajar sesuai dengan jadwal yang tepat dapat melaksanakan pembelajaran dengan tertib. Oleh karena itu, seorang guru dan juga pelajar memerlukan jadwal yang tertib dan baik agar meminimalisir terjadinya bentrok antara satu mata pelajaran dengan mata pelajaran yang lainnya [1].

Sampai saat ini, penjadwalan mata pelajaran di MI Margomulyo 02 dilakukan secara *konvensional* oleh *administrator*, dimana sebelumnya telah dilakukan rapat bersama guru kelas dan guru mapel. Dalam penyusunan jadwal, perlu dipertimbangkan beberapa faktor atau komponen seperti waktu, mata pelajaran, dan kapasitas ruangan agar terhindar dari kasus bentrok jadwal atau kosongnya kelas. Hal ini akan meminimalisir gangguan dalam proses pembelajaran dan memberikan pengalaman belajar yang optimal bagi siswa. Selain itu, ada beberapa bagian lain yang perlu diperhatikan seperti tidak ada guru yang mengajar dua jam pelajaran atau lebih pada waktu yang sama, satu ruangan hanya boleh ditempati

oleh satu guru, dan jam mengajar yang terbatas. Penyusunan jadwal yang masih dilakukan secara *konvensional* membutuhkan estimasi waktu yang tidak sedikit dan juga ketelitian, sehingga sangat memungkinkan terjadinya kesalahan dalam pembagian jadwal (*human error*). Untuk membantu administrator menyelesaikan masalah tersebut, maka peneliti akan melakukan penelitian menggunakan model *prototype* untuk metode pengembangannya dan algoritma genetika guna menyelesaikan permasalahan pada sistem penjadwalan sekolah pada MI Al-Ma'arif Margomulyo 02.

Algoritma genetika adalah salah satu teknik guna menemukan solusi optimal untuk masalah dengan banyak solusi. Teknik ini mencari di antara beberapa solusi yang diterima untuk mendapatkan solusi terbaik sesuai dengan kriteria yang telah ditentukan [2]. Algoritma genetika mempunyai keunggulan dibandingkan metode optimasi yang lain yaitu dapat mengoptimasi masalah yang ada dengan masalah yang lebih kompleks dan menyelesaikan masalah melalui langkah-langkah yang logis dan sistematis. Algoritma ini akan menghasilkan jadwal pelajaran sekolah yang lebih baik dan mempersingkat waktu. Dalam penelitian ini, peneliti menggunakan Model *prototype* yang merupakan model pengembangan sistem dengan melakukan suatu hal dengan cepat. Model ini melibatkan user pada proses desain dan user akan memberikan tanggapan berupa umpan balik pada tahap awal desain atau tahap pengembangan aplikasi [3].

Model *prototype* dapat digunakan untuk menggabungkan kurangnya pemahaman *klien* tentang pengembangan perangkat lunak pada masalah teknis

dan mengklarifikasi persyaratan yang diinginkan klien. Model *prototype* dimulai dengan mengakumulasi beberapa kebutuhan pengguna sesuai dengan sistem yang akan dikerjakan. Program *prototype* kemudian dibuat untuk membantu pelanggan memvisualisasikan dengan lebih baik apa yang sebenarnya mereka inginkan [4].

Adapun Penelitian sebelumnya oleh Indri yang mengambil judul “Pengembangan Sistem Informasi Penjadwalan Mata Kuliah Berbasis Web Di Fakultas Teknik Universitas Negeri Makassar” menggunakan metode pengembangan *SDLC* dengan model pengembangan *prototype* membuktikan bahwa pengembangan sistem penjadwalan berbasis web disini mendapatkan hasil praktis, valid, dan juga efektif. Sehingga dalam pelaksanaannya sistem yang menggunakan *prototype* pada model pengembangannya lebih maksimal dan memudahkan administrator dalam mengelola penjadwalan.

2. TINJAUAN PUSTAKA

Dalam penyusunan penelitian ini, penulis menyertakan beberapa penelitian sebelumnya yang telah dilakukan oleh pihak lain dan berkaitan dengan judul penelitian ini. Beberapa penelitian sebelumnya yang relevan dengan penelitian ini adalah sebagai berikut:

Pengembangan Model Sistem Informasi Akademik Pada Sekolah Menengah Pertama Menggunakan Metode *SDLC*. Penelitian ini mempunyai tujuan mempermudah kepala sekolah dalam memantau dan mengontrol kegiatan absensi dan informasi terkait nilai peserta didik sekolah dengan melakukan pengembangan sistem menggunakan *SDLC* [5]. Salah satu bagian dari *SDLC* yaitu model spiral yang di terapkan pada penelitian berjudul “Penerapan Model Spiral Dalam Pengembangan Sistem Informasi Penjadwalan Produksi Berbasis Website (Studi Kasus: PT. Dinar Makmur Cikarang)”. Hasil dari penelitian ini adalah proses pendistribusian jadwal produksi yang lebih cepat dan kemudahan dalam menyampaikan informasi dari bagian PPIC kepada bagian produksi [6].

Pengembangan Sistem Informasi Keberadaan Dosen Menggunakan Model *Prototype*. Model *prototype* dianggap dapat menghubungkan interaksi antara pengembang dan pengguna dalam pembuatan sistem. Dalam penelitian ini mendapatkan hasil 5 iterasi hingga aplikasi diterima oleh user [7].

Performance Algoritma Genetika (Ga) Pada Penjadwalan Mata Pelajaran. Penelitian menggunakan algoritma genetika pada kasus penjadwalan ini karena algoritma genetika dianggap memiliki kinerja yang baik dalam menemukan solusi optimal. Penelitian ini berhasil mengimplementasikan algoritma genetika pada masalah penjadwalan mata pelajaran [8]. Penelitian selanjutnya yang menggunakan algoritma genetika yaitu “Sistem Penjadwalan Praktikum Menggunakan Algoritma Genetika”. Penelitian ini

memiliki tujuan untuk mempermudah admin FTI dalam melakukan penjadwalan dan lebih efisien [9].

Pengujian Sistem Informasi *Inventory* Pada Perusahaan Distributor Farmasi Menggunakan Metode *Blackbox Testing* dengan teknik *equivalen partitioning*. Sistem ini akan mengontrol dan mengelola manajemen dan pemusnahan barang perusahaan. Pada penelitian ini, pengujian dilakukan dengan menggunakan *blackbox* pada tahap transisi dengan tujuan guna mendeteksi kesalahan yang terjadi ketika sistem digunakan oleh pengguna dengan menguji data normal dan tidak normal. *Blackbox* menguji pada bagian fungsionalitasnya terbukti mempermudah penguji perangkat lunak dan memungkinkan pengembang perangkat lunak untuk memperbaiki kesalahan karena titik-titik yang perlu diperbaiki sudah jelas dalam *ID* pengujian [10].

Pengujian *Whitebox* Pada Aplikasi *DEBT Manager* Berbasis Android. Penelitian ini melakukan penelitian dan analisis terhadap struktur internal dan kode dari perangkat lunak pada aplikasi *DEBT manager* berbasis android. Dalam pengujian menggunakan metode *whitebox* testing, setiap fungsionalitas diuji secara terpisah dengan melalui pengujian path, penilaian resiko prosedur aplikasi, dan pengujian *test case* [11].

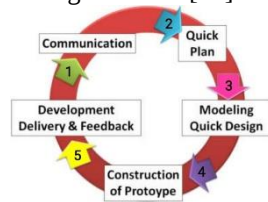
A modular fuzzy expert system for chemotherapy drug dose scheduling. Sistem penjadwalan menggunakan logika fuzzy ini digunakan untuk menentukan dosis utama berdasarkan pendapat para ahli. Strategi yang diusulkan menggunakan logika fuzzy untuk penjadwalan dosis kemoterapi dapat membantu ahli onkologi atau dokter dalam merencanakan penjadwalan pengobatan kemoterapi yang optimal [12]

An optimal construction of smart aged homes based on SDLC using smart sensors and agent networks. Dalam penelitian ini akan melakukan pengembangan berulang berdasarkan sensor pintar dan jaringan agen menggunakan model pengembangan perangkat lunak *SDLC* dengan menyinkronkan rumah lansia pintar dengan istem *cyber-fisik* industri 5.0 [13].

Dari beberapa uraian diatas, dapat di simpulkan bahwa penelitian paling mendekati dengan yang akan dilakukan adalah penelitian oleh Haifan dkk (2020) dengan judul Sistem Penjadwalan Praktikum Menggunakan Algoritma Genetika. Pada penelitian ini menggunakan algoritma genetika sebagai algoritma optimasi penjadwalan praktikum di Prodi Teknik Informatika S-1 ITN Malang, sedangkan penelitian yang akan dilakukan menggunakan metode pengembangan sistem model *prototype* dengan algoritma pendukung yaitu algoritma genetika sebagai agoritma optimasi untuk penjadwalan di MI Al-Ma'arif Margomulyo 02. Sistem yang akan di bangun berbasis web guna memudahkan pengguna dalam pengoperasiannya dan sistem di bangun untuk pembaharuan proses pembuatan jadwal.

3. METODE PENELITIAN

Metode pengembangan yang digunakan dalam penelitian ini adalah model pengembangan *prototype*, dengan tahapan sebagai berikut [14].



Gambar 1. Alur *Prototype*

1. Communication

Sebelum melakukan perencanaan, tahap awal yang dilakukan adalah komunikasi yang baik dengan melakukan wawancara secara intensif dengan pengguna.

2. Quick Plan

Pada tahap ini dilakukan perencanaan yang berfokus pada pembuatan desain layout atau tampilan yang terlihat oleh pengguna guna memudahkan pengembang dalam pembuatan aplikasi.

3. Modeling Quick Design

Pemodelan desain aplikasi menggunakan Flowchart dan Use case diagram.

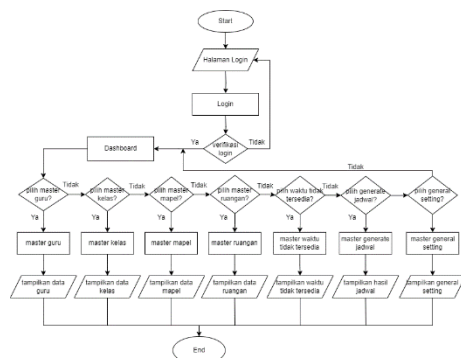
4. Contruction of Prototype

Pada tahap ini dilakukan pengkodean prototype yang mewakili sistem yang akan dibangun.

5. Development Delivery and Feedback.

Merupakan tahap akhir dari metode pengembangan prototype, sistem yang sudah jadi akan di uji dan dievaluasi oleh pengguna apakah ada yang harus diperbaiki atau sudah sesuai dengan permintaan pengguna. Pada tahap delivery melibatkan pengujian sistem menggunakan metode pengujian sistem.

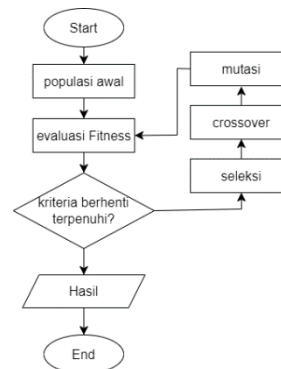
3.1. Flowchart Aplikasi



Gambar 2. Diagram Alir (Flowchart Aplikasi)

Pada gambar 2 menggambarkan diagram alir (flowchart sistem) yang merupakan flowchart aplikasi secara keseluruhan. Dimulai dengan login, selanjutnya akan masuk kehalaman dashboard dimana pengguna akan ditampilkan beberapa menu salah satunya generate jadwal.

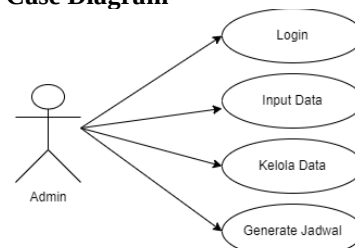
3.2. Flowchart Generate Jadwal



Gambar 3. Diagram Alir (Flowchart Generate Jadwal)

Gambar 3 menggambarkan Diagram Alir (Flowchart Generate Jadwal) yang merupakan flowchart pada proses generate jadwal. Flowchart generate jadwal menggunakan algoritma genetika sebagai algoritma optimasi untuk menghasilkan jadwal yang optimal yang diawali dengan proses populasi awal dimana terjadi pengambilan data / gen yang akan di generate. Selanjutnya akan di evaluasi apakah sudah memenuhi kriteria berhenti atau belum memenuhi. Jika sudah akan mendapatkan hasil, namun jika belum terpenuhi maka akan dilanjutkan pada proses seleksi, crossover, mutasi, evaluasi dan akan berhenti jika sudah memenuhi kriteria.

3.3. Use Case Diagram

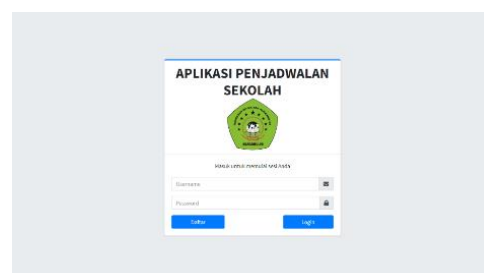


Gambar 4. Use Case Diagram

Use Case Diagram menggambarkan pengguna dapat melakukan login, input data, mengelola data dan generate jadwal. Generate jadwal akan menghasilkan jadwal yang mempermudah pengguna untuk melakukan penjadwalan.

4. HASIL DAN PEMBAHASAN

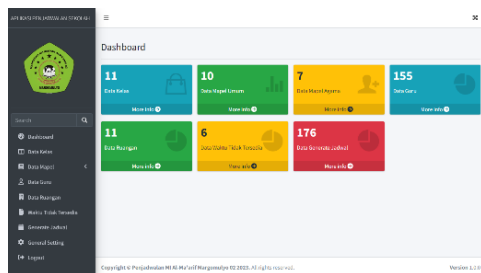
4.1. Tampilan Login



Gambar 5. Tampilan Halaman Login

Pada gambar 5 menunjukkan halaman login pengguna yang mengharuskan pengguna memasukkan *username* dan *password*.

4.2. Tampilan Halaman Dashboard Aplikasi



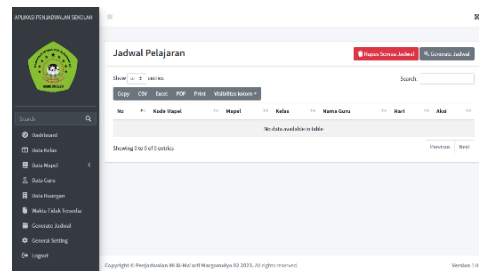
Gambar 6. Tampilan Halaman Dashboard

pada gambar 6 menggambarkan tampilan halaman menu dashboard pengguna yaitu menu data kelas, data mapel, data guru, data ruangan, waktu yang tidak tersedia, generate jadwal, general setting dan logout. Pengguna dapat melihat, mengedit dan menginput data-data yang ada di halaman *dashboard*.

Gambar 7 merupakan halaman generate jadwal dimana pengguna akan melakukan generate jadwal dan akan menghasilkan jadwal yang sudah terkomputerisasi. Pada halaman ini algoritma genetika

digunakan oleh pengembang untuk menghasilkan jadwal.

4.3. Tampilan Generate Jadwal



Gambar 7. Tampilan Halaman Generate Jadwal

4.4. Pengujian Blackbox Testing

Pengujian *blackbox testing* digunakan untuk memeriksa dan mengamati hasil input dan output dari aplikasi yang telah dikembangkan tanpa perlu mengetahui rincian struktur kode perangkat lunaknya. Dalam pengujian *blackbox testing*, teknik *equivalence partitioning* digunakan untuk memastikan bahwa kasus uji yang dihasilkan menjadi lebih akurat. Berikut ini adalah pengujian *blackbox testing* menggunakan metode *equivalence partitioning* pada aplikasi penjadwalan pembelajaran sekolah.

Table 1. Test Case Form Register

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
R01	Menginput nama "ali ridho" username dengan "Ali Ridho", password dengan "aliridho25" kemudian klik tombol register.	Register akan berhasil, sistem akan langsung mengarahkan user ke halaman dashboard	Sistem akan langsung mengarahkan user ke halaman dashboard.	Sesuai
R02	Menginput nama "ali ridho" username dengan "Ali Ridho", password dengan "ali259" kemudian klik tombol register.	Register gagal karena username sudah digunakan dan Sistem akan muncul pesan "username already in use, please try again" dan tampilan tetap di halaman register	Sistem error pada bagian MySQL muncul tulisan "18.43.28 [mysql] Status change detected:stopped18.43.28..."	Tidak sesuai
R03	Menginput nama "azizah" username dengan "azizah", password dengan "aziza" kemudian klik tombol register.	Register akan gagal dan tetap berada halaman register karena password minimal 8 karakter.	Sistem akan muncul pesan "The password must be at least 8 characters" dan tampilan tetap di halaman register.	Sesuai

Pengujian *test case form register* dilakukan sebanyak 3 kali. Hasil pengujian 3 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$1. x = \frac{2}{3} \times 100\% = 67\%$$

Keterangan:
x: presentase *test case form register*

Tabel 2. Test Case Form Login

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
L01	Menginput username menggunakan "Ali Ridho", dengan password "aliridho25" kemudian klik login	Login akan berhasil, secara otomatis sistem akan mengarahkan pengguna ke halaman dashboard.	Login berhasil dan diarahkan langsung ke halaman dashboard.	Sesuai
L02	Menginput username menggunakan "Ali Ridho" dengan password "aliridho2" lalu klik login.	Login akan gagal dan tetap berada di halaman form login.	Sistem akan muncul pesan "Periksa Kembali Username Dan Password Anda" dan tampilan tetap pada halaman form login.	Sesuai

Pengujian *test case form login* dilakukan sebanyak 2 kali. Hasil pengujian 2 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$2. x = \frac{2}{2} \times 100\% = 100\%$$

Keterangan :

x : persentase *test case form login*

Tabel 3. Test Case Form Data Kelas

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
K01	Menginput tambah data kelas dengan "6A", jumlah siswa 20 dan klik submit.	Data akan tersimpan ke database dan data yang tersimpan akan ditampilkan di halaman menu data kelas	Data berhasil disimpan dan data yang tersimpan ditampilkan di halaman menu data kelas.	Sesuai
K02	Mengedit data kelas "6A" dari jumlah siswa 20 menjadi 23. Klik submit.	Data jumlah siswa akan berubah menjadi 23 siswa dan tersimpan di database.	Data berhasil berubah di menu data kelas dan juga di database.	Sesuai
K03	Menghapus data kelas "6A" dengan jumlah siswa 23. Klik OK	sistem akan mengirim pesan "Are you sure you want to delete this data?" klik OK dan Data akan terhapus dari tampilan menu data kelas dan di database.	Data terhapus dari tampilan menu data kelas dan di database.	Sesuai

Pengujian *test case form data kelas* dilakukan sebanyak 3 kali dengan perhitungan persentase sesuai rumus berikut.

$$3. x = \frac{3}{3} \times 100\% = 100\%$$

Keterangan :

x : persentase *test case form data kelas*

Tabel 4. Test Case Form Mapel Umum

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
U01	Menginput tambah data mapel umum dengan kode "UM11", mapel "bahasa jepang", durasi 60 dan klik submit.	Data akan tersimpan ke database dan data yang tersimpan akan ditampilkan di halaman menu data mapel umum	Data berhasil disimpan dan data yang tersimpan ditampilkan di halaman menu data mapel umum.	Sesuai
U02	Mengedit data mapel umum dengan kode "UM11", mapel "bahasa Cina", durasi 60 dan klik submit.	Data mapel akan berubah menjadi bahasa Cina dan tersimpan di database.	Data berhasil berubah di menu data mapel umum dan juga di database.	Sesuai
U03	Menghapus data mapel umum dengan kode "UM11", mapel "bahasa Cina", durasi 60 dan klik OK	sistem akan mengirim pesan "Are you sure you want to delete this data?" klik OK dan Data akan terhapus dari tampilan menu data mapel umum dan di database.	Data terhapus dari tampilan menu data mapel umum dan di database.	Sesuai

Pengujian *test case form data mapel umum* dilakukan sebanyak 3 kali. Hasil pengujian 3 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$4. x = \frac{3}{3} \times 100\% = 100\%$$

Keterangan :

x : persentase *test case form data mapel umum*

Tabel 5. Test Case Mapel Agama

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
A01	Menginput tambah data mapel agama dengan kode "AG8", mapel "Tafsir", durasi 30 dan klik submit.	Data akan tersimpan ke database dan data yang tersimpan akan ditampilkan di halaman menu data mapel agama	Data berhasil disimpan dan data yang tersimpan ditampilkan di halaman menu data mapel agama	Sesuai
A02	Mengedit data mapel agama dengan kode "AG8", mapel "Tafsir Ibrisi" durasi 30 dan klik submit.	Data mapel akan berubah menjadi Tafsir Ibrisi dan tersimpan di database.	Data berhasil berubah di menu data mapel agama dan juga di database.	Sesuai
A03	Menghapus data mapel agama dengan kode "AG8", mapel "Tafsir Ibrisi" durasi 30 dan klik OK	sistem akan mengirim pesan "Are you sure you want to delete this data?" klik OK dan Data akan terhapus dari tampilan menu data mapel agama dan di database.	Data terhapus dari tampilan menu data mapel agama dan di database.	Sesuai

Pengujian *test case form* data mapel agama dilakukan sebanyak 3 kali. Hasil pengujian 4 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$5. x = \frac{3}{3} \times 100\% = 100\%$$

Keterangan:

x : persentase *test case form* data mapel agama

Tabel 6. Test Case Form Data Guru

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
G01	Menginput tambah data guru dengan nama guru "diny ari", jabatan "Guru" mapel "IPA", no hp "085822362736" dan klik submit.	Data akan tersimpan ke database dan data yang tersimpan akan di tampilkan di halaman menu data guru	Data berhasil disimpan dan data yang tersimpan ditampilkan di halaman menu data guru	Sesuai
G02	Mengedit data guru dengan nama guru "diny ari bintary", jabatan "Guru" mapel "IPA", no hp "085822362736" dan klik submit.	Data guru akan berubah menjadi diny ari bintary dan tersimpan di database.	Data berhasil berubah di menu data guru dan juga di database.	Sesuai
G03	Menghapus data guru dengan nama guru "diny ari bintary", jabatan "Guru" mapel "IPA", no hp "085822362736" dan klik OK	sistem akan mengirim pesan "Are you sure you want to delete this data?" klik OK dan Data akan terhapus dari tampilan menu data guru dan di database.	Data terhapus dari tampilan menu data guru dan di database.	Sesuai

Pengujian *test case form* data guru dilakukan sebanyak 3 kali. Hasil pengujian 3 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$= 100\%$$

Keterangan :

x : persentase *test case form* data guru

$$6. x = \frac{3}{3} \times 100\%$$

Tabel 7. Test Case Form Data Ruangan

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
R01	Menginput tambah data ruangan dengan "12", dan klik submit.	Data akan tersimpan ke database dan data yang tersimpan akan di tampilkan di halaman menu data ruangan	Data berhasil disimpan dan data yang tersimpan ditampilkan di halaman menu data ruangan	Sesuai
R02	Mengedit data ruangan dengan "13" klik submit	Data guru akan berubah menjadi 13 dan tersimpan di database.	Data berhasil berubah di menu data ruangan dan juga di database.	Sesuai
R03	Menghapus data ruangan dengan "12" klik OK	sistem akan mengirim pesan "Are you sure you want to delete this data?" klik OK dan Data akan terhapus dari tampilan menu data ruangan dan di database.	Data terhapus dari tampilan menu data ruangan dan di database.	Sesuai

Pengujian *test case form* data ruangan dilakukan sebanyak 3 kali. Hasil pengujian 3 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$7. x = \frac{3}{3} \times 100\% = 100\%$$

Keterangan :

x : persentase *test case form* data ruangan

Tabel 8. Test Case Form Waktu Tidak Tersedia

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
W01	Menginput tambah data dengan "sabtu", waktu tidak tersedia "10.00-10.30" dan klik submit.	Data akan tersimpan ke database dan data yang tersimpan akan di tampilkan di halaman menu data waktu tidak tersedia	Data berhasil disimpan dan data yang tersimpan ditampilkan di halaman menu data waktu tidak tersedia.	Sesuai
W02	Mengedit data dengan "sabtu", waktu tidak tersedia "10.00-10.15" dan klik submit.	Data waktu tidak tersedia akan berubah menjadi "10.00-10.15" dan tersimpan di database.	Data berhasil berubah di menu data waktu tidak tersedia dan juga di database.	Sesuai
W03	Menghapus data "sabtu", waktu tidak tersedia "10.00-10.30" dan klik OK	sistem akan mengirim pesan "Are you sure you want to delete this data?" klik OK dan Data akan terhapus dari tampilan menu data waktu tidak tersedia dan di database.	Data terhapus dari tampilan menu data waktu tidak tersedia dan di database.	Sesuai

Pengujian *test case form* waktu tidak tersedia dilakukan sebanyak 3 kali. Hasil pengujian 3 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$8. x = \frac{3}{3} \times 100\% = 100\%$$

Keterangan :

x : presentase *test case form* waktu tidak tersedia

Tabel 9. Test Case Form Generate Jadwal

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
J01	Klik generate jadwal	Data akan teracak sesuai dengan aturan algoritma genetika dan hasil akan tampil di halaman generate jadwal.	Data berhasil tergenerate, data akan tersimpan dan ditampilkan di halaman menu generate jadwal	Sesuai
J02	Menghapus data jadwal hasil generate dengan Klik Hapus semua jadwal	Data akan terhapus dari tampilan menu generate jadwal dan juga terhapus di database.	Data terhapus dari tampilan menu generate jadwal dan di database.	Sesuai

Pengujian *test case form generate jadwal* dilakukan sebanyak 2 kali. Hasil pengujian 2 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$9. x = \frac{2}{2} \times 100\%$$

$$= 100\%$$

Keterangan :

x : presentase *test case form generate jadwal*

Tabel 10. Test Case Form General Setting

Id	Langkah Pengujian	Hasil yang Diinginkan	Hasil dari Pengujian	Kesimpulan
S01	Menginput data mapel agama "3" mapel umum "5" klik submit	Data akan tersimpan kedalam database.	Data berhasil tersimpan di database.	Sesuai

Pengujian *test case form general setting* dilakukan sebanyak 1 kali yaitu menginput data mapel umum dan agama, sistem akan menyimpan di database. Hasil pengujian 1 fungsi tersebut dihitung persentasenya sesuai rumus sebagai berikut.

$$10. x = \frac{1}{1} \times 100\% = 100\%$$

Keterangan :

x : presentase *test case form general setting*

Hasil keseluruhan dari presentase hasil pengujian *test case form register, form login, form data kelas, data guru, data mapel umum, data mapel agama, data ruangan, data waktu tidak tersedia, generate dan general setting* dapat dihitung menggunakan rumus dibawah ini.

$$11. x = \frac{967}{10} \times 100 = 96,7\%$$

Keterangan :

x : presentase hasil keseluruhan

Jadi hasil persentase *blackbox testing* secara keseluruhan adalah sebesar 96,7%.

4.5. Whitebox Testing

Metode pengujian *whitebox testing* digunakan untuk memeriksa dan mengamati sebuah metode pada perangkat lunak tanpa perlu memperhatikan detail internal perangkat lunak. Berikut ini adalah langkah-langkah pengujian *whitebox testing* pada aplikasi penjadwalan pembelajaran sekolah.

4.6. Pemetaan Source Code

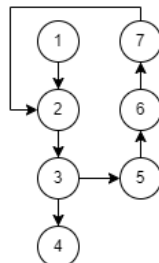
Tabel 11. Tabel Pemetaan Source Code

Node	Source Code
1	DB::table('generate')->insert(['kode_mapel' => \$request->kode_mapel, 'mapel' => \$nama_mapel, 'kelas' => \$request->kelas, 'nama_guru' => \$request->nama_guru, 'hari' => \$request->hari, 'waktu' => \$request->waktu,]);
2	\$mapel = []; \$limit_mapel_agama = (int)generalSetting('mapel_agama')->value; \$limit_mapel_umum = (int)generalSetting('mapel_umum')->value; foreach(\$mapel_agama as \$key => \$value) { \$dataGuru = DataGuru::where('code_mapel', \$value->kode_agama) ->select('nama_guru') ->first(); \$value->nama_guru = \$dataGuru->nama_guru ?? null; if (\$value->kode_agama && \$limit_mapel_agama >= \$key+1) { array_push(\$mapel, \$value); } }
3	foreach(\$mapel_umum as \$key => \$value) { \$dataGuru = DataGuru::where('code_mapel', \$value->kode_umum) ->select('nama_guru') ->first(); \$value->nama_guru = \$dataGuru->nama_guru ?? null; if (\$dataGuru->nama_guru && \$limit_mapel_umum >= \$key+1) { array_push(\$mapel, \$value); } }
4	\$generate_waktu = \$this->generateWaktu(\$mapel); \$result_generate = \$this->generateKelas(\$generate_waktu); foreach (\$result_generate as \$key => \$data) { \$create = Generate::create(\$data); }
5	\$data[\$no] = ['hari' => \$day, 'waktu' => \$waktu_kbm, 'number_day' => \$key+1,

Node	Source Code
	<pre> 'number_sorting' => \$sorting, 'kode_mapel' => \$value->kode_umum ?? \$value->kode_agama, 'mapel' => \$value->mapel, 'durasi' => \$value->durasi, 'kelas' => null, 'ruang' => null, 'nama_guru' => \$value->nama_guru,]; \$waktu_mulai = \$waktu_mulai- >addMinutes(\$value->durasi); \$no++; \$sorting++; } </pre>
6	<pre> \$dataGenerate = []; \$count_result_generate_kelas = count(\$result_generate_kelas) - 1; \$index_result_generate_kelas = 0; </pre>
7	<pre> foreach (\$result_generate_kelas as \$key => \$value) { foreach (\$data as \$key => \$detail) { \$result = ['hari' => \$detail['hari'] ?? 'err', 'waktu' => \$detail['waktu'] ?? 'err', 'kode_mapel' => \$detail['kode_mapel'] ?? 'err', 'mapel' => \$detail['mapel'] ?? 'err', 'durasi' => \$detail['durasi'] ?? 'err', 'nama_guru' => \$detail['nama_guru'] ?? 'err', 'kelas' => \$value['kelas'] ?? 'err', 'ruang' => \$value['ruang'] ?? 'err',]; } } </pre>

4.7. Membuat Flowgraph

Pembuatan *flowgraph* digunakan untuk menggambarkan aliran kontrol logis dalam sistem. Simbol berbentuk lingkaran digunakan untuk merepresentasikan pernyataan kode program. Di bawah ini terdapat *flowgraph* untuk aplikasi penjadwalan pembelajaran sekolah menggunakan algoritma genetika.



Gambar 8. Flowgraph Generate Jadwal

Keterangan :

- 1 : Populasi Awal
- 2 : Evaluasi Fitness
- 3 : Kriteria Berhenti Terpenuhi
- 4 : Hasil
- 5 : Seleksi
- 6 : Crossover
- 7 : Mutasi

4.8. Perhitungan Cyclomatic Complexity

Perhitungan *Cyclomatic complexity* digunakan untuk mengevaluasi tingkat kompleksitas sebuah program berdasarkan logika yang telah diterapkan di dalamnya. Dalam konteks pengujian perangkat lunak, *Cyclomatic complexity* dapat digunakan untuk menentukan jumlah minimum kasus uji yang perlu dijalankan untuk menguji program menggunakan teknik *basis path testing*. Berikut ini adalah nilai *Cyclomatic complexity* untuk aplikasi penjadwalan pembelajaran sekolah menggunakan algoritma genetika.

$$12. V(G) = (7 - 7) + 2 = 2$$

Keterangan :

$V(G)$: nilai *Cyclomatic complexity*

Hasil perhitungan *cyclomatic complexity* yang diperoleh sebesar 2 dengan kompleksitas dan resiko termasuk *type of procedure* dari *a simple procedure* dengan resiko yaitu low.

4.9. Penentuan Independent Path

Penentuan *independent path* sebagai rangkaian dari proses pengujian, *independent path* merujuk dari hasil perhitungan *cyclomatic complexity* yang telah ditentukan. Berikut dibawah ini *independent path* pada aplikasi penjadwalan pembelajaran sekolah menggunakan algoritma genetika.

Tabel 12. Tabel Independent Path

Basis Flow	Independent Path
1	1-2-3-4
2	1-2-3-5-6-7-3-4

Jadi penentuan *independent path* memiliki *basis flow* sebanyak 2 dengan urutan *independent path* 1 – 2 – 3 – 4 dan 1-2-3-5-6-7-3-4.

4.10. Beta Testing

Pengujian beta testing dilakukan secara langsung terhadap pengguna yaitu bagian kurikulum dan guru dengan 18 pertanyaan. Pertanyaan tersebut berupa kuisioner mengenai kepuasan pengguna terhadap aplikasi yang telah di bangun dan di buat. Berikut dibawah ini data hasil kuisioner pengguna untuk aplikasi penjadwalan pembelajaran sekolah.

Tabel 13. Tabel Data Hasil Kuisioner Pengguna

No	Pertanyaan / Pernyataan	Penilaian			
A.	Pertanyaan/ Pernyataan dari aspek desain sistem	SB	B	C	K
1	bagaimana pendapat anda terhadap tampilan antarmuka aplikasi penjadwalan pembelajaran sekolah ini?	0	2	0	0
2	bagaimana pendapat anda mengenai fitur generate pada aplikasi ini?	2	0	0	0
3	bagaimana pendapat anda terkait tampilan menu pada aplikasi ini ?	0	1	1	0
4	bagaimana pendapat anda mengenai penggunaan kata dan bahasa pada aplikasi ini?	0	2	0	0
5	bagaimana pendapat anda mengenai tampilan awal aplikasi ini?	0	2	0	0
6	bagaimana pendapat anda mengenai tata letak menu-menu pada aplikasi ini?	1	1	0	0
B.	Pertanyaan/ Pernyataan dari aspek kesesuaian sistem	SB	B	C	K
1	aplikasi sudah berjalan sesuai dengan algoritma genetika	2	0	0	0

No	Pertanyaan / Pernyataan	Penilaian			
2	fitur-fitur yang tersedia sudah sesuai dengan kebutuhan pengguna	0	2	0	0
3	data yang dihasilkan dari aplikasi ini sudah sesuai dan benar	1	0	1	0
4	menu yang ada sudah berfungsi dengan baik dan benar	1	1	0	0
5	proses penginputan data, update / edit data, dan view data sudah berfungsi dengan baik	1	1	0	0
6	informasi yang di tampilkan sudah sesuai dengan yang di inginkan	0	2	0	0
C. Pertanyaan/ Pernyataan dari aspek kemudahan sistem		SB	B	C	K
1	aplikasi mudah digunakan	2	0	0	0
2	aplikasi mudah diakses	1	1	0	0
3	halaman register dan login dapat diakses dengan mudah	0	1	1	0
4	fitur-fitur yang ada mudah diakses	0	2	0	0
5	Proses pengolahan data seperti penginputan data, pembaruan, dan tampilan data dirancang dengan antarmuka yang mudah dipahami dan digunakan.	1	1	0	0
6	hasil data yang diperoleh dari fitur generate dapat dipahami kegunaanya	1	1	0	0
JUMLAH TOTAL		13	20	3	0

Tabel 13 berisi jumlah hasil data kuisiooner pengguna yang akan dihitung total keseluruhan nilai dari pertanyaan yang telah diisi oleh pengguna. Jumlah total dari tabel diatas digunakan untuk menghitung

presentase dari aplikasi yang telah dibuat, maka hasil tabulasi dapat ditunjukkan pada tabel 14.

Tabel 14. Tabel Hasil Tabulasi

Nilai Penilaian	Jumlah Penilaian	Total (Skala * Jumlah)
1	0	0
2	3	6
3	20	60
4	13	52
Total Skor		118
Skor Maksimum		$2 * 18 * 4 = 144$ (Jumlah Ahli IT * Jumlah Butir Soal * Skala Tertinggi)

$$\text{presentase kelayakan} = \frac{981}{1224} \times 100\% = 80.1\%$$

Setelah melakukan pengolahan data dari kuisioner yang diisi oleh responden, diperoleh hasil perhitungan keseluruhan sebesar 80.1%. Dengan demikian, dapat disimpulkan bahwa sistem termasuk dalam kategori "sangat bagus" sesuai dengan kategori berdasarkan skala interval.

5. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang dilakukan peneliti, maka peneliti mengambil beberapa kesimpulan antara lain: Pembuatan sistem aplikasi ini menggunakan *framework laravel* dan *database phpMyAdmin* yang di dalam aplikasinya menampilkan halaman register, halaman login, halaman dashboard, halaman menu kelas, halaman menu mapel agama, halaman mapel umum, halaman guru, halaman ruangan, halaman waktu tidak tersedia, halaman general setting dan halaman generate jadwal. Pada halaman generate jadwal menerapkan algoritma genetika, sehingga pengguna dapat melakukan penjadwalan secara terkomputerisasi. Hasil Pengujian aplikasi mendapatkan hasil, yaitu pada pengujian *black box* testing sebesar 96,7%, pengujian *white box* testing dengan kompleksitas dan risiko termasuk tipe *prosedur* dari *a simple procedure* dengan resiko yaitu *low*. *Beta testing* yang dilakukan oleh pengguna

dengan jumlah persentase angket pengujian mendapatkan hasil sebesar 80.1%. Dalam penelitian ini, terdapat berbagai keterbatasan yang perlu diperbarui dan diperbaiki. Sebagai saran untuk pengembangan aplikasi ini di masa mendatang, beberapa hal berikut dapat dipertimbangkan: penambahan fitur seperti integrasi dengan kalender akademik sekolah. Peningkatan antarmuka pengguna, penambahan kriteria misalnya dapat memenuhi permintaan guru untuk mengajar pada jam tertentu. Hasil penjadwalan berupa jadwal yang sudah tertata perhari dalam 1 minggu sehingga mempermudah guru dalam melihat jadwal.

DAFTAR PUSTAKA

- [1] Ana, Y., & Alaeyda, V. N. (2021). Implementasi Metode Algoritma Genetika Untuk Menentukan Penjadwalan Pelajaran Di Smkn 1. *Jurnal Ilmiah Ilkominfo - Ilmu Komputer & Informatika*, 4(1).<https://doi.org/10.47324/Ilkominfo.V4i1.114>
- [2] Sari, D. D. P., Mahmudy, W. F., & Ratnawati, D. E. (2015). Optimasi Penjadwalan Mata Pelajaran Menggunakan Algoritma Genetika (Studi Kasus: Smpn 1 Gondang Mojokerto). *Universitas Brawijaya*, 5(13).
- [3] Lutfiani, A., & Ziveria, M. (2022). *Aplikasi Penjadwalan Dan Presensi Sekolah Berbasis Web Pada Smk Kemala Bhayangkari 1 Jakarta*. 8(3), 3272–3288.

- [4] Anugrah, I. (2018). Pengembangan Sistem Informasi Penjadwalan Mata Kuliah Berbasis Web Di Fakultas Teknik Universitas Negeri Makassar. *Jurnal Pendidikan*, 6(2).
- [5] Sandi, S. (2020). Pengembangan Model Sistem Informasi Akademik Pada Sekolah Menengah Pertama Menggunakan Metode Sdlc. *Jutisi : Jurnal Ilmiah Teknik Informatika Dan Sistem Informasi*, 9(2), 41. <https://doi.org/10.35889/jutisi.V9i2.514>
- [6] Wahyuni, S., & Cahyani, N. (2020). Penerapan Model Spiral Dalam Pengembangan Sistem Informasi Penjadwalan Produksi Berbasis Website (Studi Kasus: Pt. Dinar Makmur Cikarang). *Informatics And Digital Expert (Index)*, 2(1), 1–6. <https://doi.org/10.36423/ide.V2i1.425>
- [7] Wahyuni, E. D. (2021). Pengembangan Sistem Informasi Keberadaan Dosen Menggunakan Model Prototype. *Jurnal Tekno Kompak*, 15(2). <https://doi.org/10.33365/jtk.V15i2.1135>
- [8] M.Kom., E. D. (2016). Performance Algoritma Genetika (Ga) Pada Penjadwalan Mata Pelajaran. *Infotekjar (Jurnal Nasional Informatika Dan Teknologi Jaringan)*, 1(1). <https://doi.org/10.30743/infotekjar.V1i1.42>
- [9] Haifan, A., Agus Pranoto, Y., & Vendyansyah, N. (2020). Sistem Penjadwalan Praktikum Menggunakan Algoritma Genetika (Studi Kasus : Teknik Informatika S-1 Itn Malang). *Jati (Jurnal Mahasiswa Teknik Informatika)*, 4(1). <https://doi.org/10.36040/jati.V4i1.2273>
- [10] Setiyani, L. (2019). Pengujian Sistem Informasi Inventory Pada Perusahaan Distributor Farmasi Menggunakan Metode Black Box Testing. *Techno Xplore : Jurnal Ilmu Komputer Dan Teknologi Informasi*, 4(1). <https://doi.org/10.36805/technoxplore.V4i1.539>
- [11] Gusdevi, H., Kuswayati, S., Iqbal, M., Abu Bakar, M. F., Novianti, N., & Ramadan, R. (2022). Pengujian White-Box Pada Aplikasi Debt Manager Berbasis Android. *Naratif: Jurnal Nasional Riset, Aplikasi Dan Teknik Informatika*, 4(1), 11–22. <https://doi.org/10.53580/naratif.V4i1.147>
- [12] Faisal, R. H., Debnath, S., Islam, M. M. U., Sifath, S., Kakon, S. A., Alam, M. S., & Siddique, N. (2023). A Modular Fuzzy Expert System For Chemotherapy Drug Dose Scheduling. *Healthcare Analytics*, 3(November 2022), 100139. <https://doi.org/10.1016/j.health.2023.100139>
- [13] Subha, R., & Zhang, J. (2022). An Optimal Construction Of Smart Aged Homes Based On Sdlc Using Smart Sensors And Agent Networks. *International Journal Of Intelligent Networks*, 3(September), 138–142. <https://doi.org/10.1016/j.ijin.2022.09.003>
- [14] Pinto, M. D. R., Widodo, W., & Rachman, A. (2020). Rancang Bangun Aplikasi Pemesanan Air Bersih Berbasis Android Dengan Menggunakan Model Prototype. *Integer: Journal Of Information Technology*, 5(1). <https://doi.org/10.31284/j.integer.2020.V5i1.905>