

Prototipe Interaktif Pengolahan Nilai Mahasiswa dengan Visual Basic: Pendekatan Desain Berpusat pada Pengguna.

Muhammad Anwar Hawariy¹, Fitran Amin Lailan¹, Dellika Saltariksa¹, Dyah Wahyuningrum¹, Siti Shahira Azzahra¹, Abdillah Haidar Nasyaev², Ameliyana Rizky Syamara Putri Akhmad Yani² dan Hasyim Asyari^{1*}

¹Program Studi Teknik Industri, Universitas Jenderal Soedirman,
Purbalingga, Jawa Tengah, Indonesia

² Program Studi Teknik Mesin, Universitas Jenderal Soedirman,
Purbalingga, Jawa Tengah, Indonesia

*E-mail: Hasyim.asyari@unsoed.ac.id (corresponding author)

Abstrak

Penelitian ini bertujuan untuk mengembangkan prototipe aplikasi pengolahan nilai mahasiswa berbasis desktop menggunakan bahasa pemrograman Visual Basic .NET (VB.NET) pada platform Visual Studio. Latar belakang penelitian ini didasari oleh kebutuhan akan sistem pengelolaan data akademik yang lebih efisien dan akurat, khususnya dalam proses penginputan dan pengolahan nilai mahasiswa yang selama ini seringkali dilakukan secara manual. Prototipe ini dirancang untuk memfasilitasi perhitungan nilai akhir secara otomatis berdasarkan komponen penilaian seperti Tugas dan Absensi, Ujian Tengah Semester, dan Ujian Akhir Semester. Metodologi penelitian yang digunakan meliputi studi literatur terkait sistem informasi akademik dan penggunaan Visual Basic dalam pengembangan aplikasi desktop, perancangan diagram alir program untuk memvisualisasikan alur kerja aplikasi, pengembangan antarmuka pengguna (user interface) yang sederhana dan intuitif dengan komponen seperti Label, TextBox, Button, dan DataGridView, serta implementasi logika program menggunakan kode VB.NET untuk input data, pemrosesan nilai, penyimpanan, penghapusan, dan tampilan hasil. Pengujian prototipe dilakukan dengan memasukkan data mahasiswa dan nilai untuk memverifikasi fungsionalitas aplikasi. Hasil pengembangan berupa prototipe aplikasi yang mampu melakukan penginputan data nilai, perhitungan nilai akhir secara otomatis, dan menampilkan daftar nilai mahasiswa. Desain antarmuka yang sederhana dan pengelompokan input output yang jelas bertujuan untuk meningkatkan kemudahan penggunaan bagi dosen atau operator. Kesimpulan dari penelitian ini menunjukkan bahwa pemanfaatan Visual Basic .NET dalam pengembangan prototipe aplikasi pengolahan nilai mahasiswa menawarkan solusi yang cepat, tepat, dan efisien, serta berpotensi mengurangi risiko kesalahan perhitungan manual dan mempermudah pengelolaan data akademik. Pengembangan lebih lanjut dapat dilakukan dengan menambahkan fitur validasi data yang lebih komprehensif dan integrasi dengan basis data yang lebih kompleks..

Kata kunci: Algoritma, Nilai, Pemograman, Visual basic

1. Pendahuluan

Di era digital yang serba terhubung ini, pengelolaan data telah bertransformasi menjadi fondasi krusial dalam berbagai sektor, tak terkecuali dalam ekosistem pendidikan tinggi. Data nilai mahasiswa, sebagai representasi capaian akademik individu, memegang peranan sentral dalam proses evaluasi pembelajaran dan pengambilan keputusan strategis di tingkat institusi [1]. Proses penginputan dan pengolahan data nilai yang efisien dan akurat bukan hanya memfasilitasi dosen dalam memberikan penilaian yang objektif dan tepat waktu [2], tetapi juga memberdayakan mahasiswa untuk secara proaktif memantau kemajuan akademis mereka, mendorong keterlibatan dan refleksi diri dalam pembelajaran [3]. Sebaliknya, metode pengelolaan nilai yang manual dan terfragmentasi berpotensi menimbulkan inefisiensi, kesalahan input, serta keterlambatan dalam penyediaan informasi yang dibutuhkan oleh berbagai pihak [4]. Oleh karena itu, implementasi sistem informasi yang terstruktur dan mudah digunakan menjadi sebuah keniscayaan untuk meningkatkan kualitas pengelolaan data nilai mahasiswa.

Sejalan dengan kebutuhan akan solusi praktis dan cepat dalam pengembangan aplikasi, Visual Basic telah menjadi pilihan yang relevan dalam berbagai penelitian di Indonesia, termasuk dalam konteks sistem informasi pendidikan. Beberapa studi dalam jurnal nasional menunjukkan pemanfaatan Visual Basic untuk mengembangkan aplikasi berbasis desktop yang fokus pada kebutuhan spesifik institusi. Misalnya, penelitian lain melakukan eksplorasi pengembangan

sistem informasi perpustakaan sekolah menggunakan Visual Basic, yang menekankan pada antarmuka pengguna yang intuitif bagi pustakawan [5]. Studi lain oleh [6] merancang aplikasi pengelolaan data siswa dengan fitur pelaporan yang sederhana menggunakan platform yang sama. Lebih lanjut, [7] dalam penelitiannya mengimplementasikan Visual Basic untuk membangun sistem pendukung keputusan sederhana dalam penjadwalan perkuliahan. Visual basic digunakan pula dalam aplikasi pembelajaran kalkulator konversi suhu [8]. Riset-riset ini umumnya menyoroti kemudahan penggunaan Visual Basic dalam menghasilkan prototipe atau aplikasi dengan fokus pada fungsionalitas dasar dan antarmuka pengguna yang familiar bagi pengguna Windows. Meskipun demikian, penting untuk dicatat bahwa konteks dan fokus penelitian yang menggunakan Visual Basic dalam jurnal nasional cenderung pada solusi skala kecil atau prototipe untuk mengatasi permasalahan spesifik dalam lingkup institusi atau organisasi tertentu.

Sistem manual seringkali memicu kebutuhan akan solusi digital yang terintegrasi. Penelitian yang telah dilakukan mengenai pengembangan prototipe sistem pengolahan nilai mahasiswa menggunakan bahasa pemrograman Visual Basic. Tujuan utama dari pengembangan prototipe ini adalah untuk memberikan gambaran konkret mengenai fungsionalitas dasar sebuah sistem pengolahan nilai yang mudah digunakan dan berpotensi untuk dikembangkan lebih lanjut. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi bagi pemahaman tentang bagaimana teknologi sederhana namun efektif dapat diimplementasikan untuk menjawab kebutuhan spesifik dalam administrasi akademik, serta membuka peluang untuk pengembangan sistem yang lebih canggih di masa depan.

2. Metodologi

Tahapan awal dalam pengembangan prototipe ini adalah "buka aplikasi Visual Studio di komputer", kemudian dipilih opsi "Create a new project". Langkah selanjutnya adalah memilih template proyek "Windows Forms App (.NET Framework)" apabila lingkungan pengembangan yang digunakan adalah Windows Forms berbasis .NET Framework. Proyek kemudian diberi nama, sebagai contoh "InputNilaiMahasiswa", diikuti dengan penekanan tombol "Create" untuk inisiasi proyek baru. Setelah jendela form utama terbuka, proses dilanjutkan dengan "mulai mendesain tampilan aplikasi dengan menambahkan beberapa komponen dari Toolbox, seperti Label, TextBox, Button, dan DataGridView". Fungsi dari setiap komponen adalah sebagai berikut: "Label digunakan untuk memberikan keterangan pada setiap kolom input, seperti Nama, NIM, Mata Kuliah, dan Nilai"; "TextBox digunakan untuk menginput data tersebut"; ditambahkan pula "tiga buah Button yang berfungsi untuk menyimpan data, menghapus data (clear), dan keluar dari aplikasi"; dan "terakhir, tambahkan DataGridView untuk menampilkan daftar data mahasiswa yang telah dimasukkan".

Setelah perancangan antarmuka selesai, langkah berikutnya adalah "atur properti masing-masing komponen agar mudah dikenali dalam penulisan kode". Sebagai ilustrasi, "beri nama TextBox dengan nama yang sesuai seperti txtNama, txtNIM, txtMatkul, dan txtNilai". Demikian pula untuk tombol, diberikan label visual berupa teks seperti "Simpan", "Clear", dan "Keluar" "agar pengguna tahu fungsi masing-masing tombol". Implementasi logika program dimulai dengan "klik ganda pada tombol 'Simpan' untuk menambahkan kode program". Kode yang dimasukkan berfungsi untuk "mengambil data dari TextBox dan menambahkannya ke dalam baris DataGridView". Sebagai contoh implementasi, dapat "membuat array string berisi data dari setiap TextBox, lalu menambahkannya ke DataGridView menggunakan Rows.Add".

Untuk fungsionalitas tombol "Clear", ditambahkan "kode yang akan mengosongkan semua TextBox dan mengarahkan fokus kembali ke TextBoxNIM". Sementara itu, pada tombol "Keluar", dimasukkan "perintah End untuk menutup aplikasi saat tombol ditekan". Proses pengujian prototipe dilakukan setelah penulisan kode selesai, yaitu dengan "jalankan aplikasi dengan menekan tombol Start atau F5 pada keyboard". Pengujian awal dilakukan dengan "coba dengan mengisi data mahasiswa dan klik tombol Simpan untuk melihat hasilnya pada DataGridView". Sebagai pengembangan lebih lanjut,

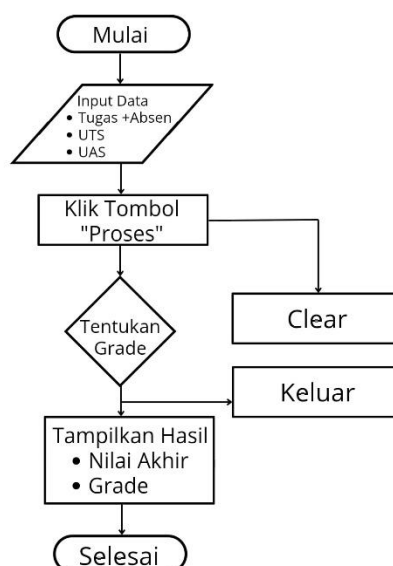
"dapat juga menambahkan validasi agar tidak terjadi input kosong, misalnya dengan membuat pengecekan jika TextBox masih kosong dan menampilkan pesan peringatan".

2.1 Algoritma Pemograman

Algoritma sebagai deskripsi abstrak dan formal dari prosedur komputasi [9]. Algoritma pemograman yang dirancang pada program yang telah dibuat adalah sebagai berikut :

1. Mulai
2. Input Data Mahasiswa
 - Nilai Tugas + Absen
 - Nilai UTS
 - Nilai UAS
3. Proses Hitung Nilai Akhir
$$\text{Nilai Akhir} = (0.2 \times \text{Tugas}) + (0.3 \times \text{UTS}) + (0.5 \times \text{UAS})$$
4. Keputusan (Decision)
 - Jika Nilai Akhir $\geq 85 \rightarrow \text{Grade} = \text{A}$
 - Jika Nilai Akhir $\geq 75 \rightarrow \text{Grade} = \text{B}$
 - Jika Nilai Akhir $\geq 65 \rightarrow \text{Grade} = \text{C}$
 - Jika Nilai Akhir $\geq 50 \rightarrow \text{Grade} = \text{D}$
 - Jika $< 50 \rightarrow \text{Grade} = \text{E}$
5. Tampilkan Nilai Akhir dan Grade
6. Pilihan
 - Klik "Proses" $\rightarrow$ kembali ke perhitungan
 - Klik "Clear" $\rightarrow$ kembali ke input kosong
 - Klik "Keluar" $\rightarrow$ selesai
7. Selesai

2.2 Diagram Alir pemograman



Gambar 1. Diagram Alir Pemograman

Diagram alir adalah representasi grafis dari suatu proses atau algoritma, yang menggambarkan langkah-langkah secara berurutan menggunakan simbol-simbol standar [10]. Diagram alir program yang ditunjukkan pada gambar 1 menunjukkan alur kerja sebuah aplikasi sederhana untuk mengolah nilai mahasiswa. Proses dimulai dengan tindakan "Mulai", yang mengindikasikan awal dari eksekusi program. Selanjutnya, pengguna diminta untuk melakukan "Input Data" yang terdiri dari beberapa komponen penilaian, yaitu Tugas dan Absensi, Ujian Tengah Semester (UTS), dan Ujian Akhir Semester (UAS). Setelah data nilai dimasukkan, pengguna perlu melakukan aksi "Klik Tombol 'Proses'" untuk menginstruksikan program agar memulai perhitungan.

Program kemudian akan masuk ke tahap "Tentukan Grade", yang merupakan proses logika untuk menentukan nilai akhir dan menetapkan grade berdasarkan kriteria penilaian yang telah ditentukan. Setelah grade berhasil ditentukan, program akan "Tampilkan Hasil" yang meliputi nilai akhir mahasiswa dan grade yang bersangkutan. Alur program kemudian berakhir pada "Selesai". Selain alur utama pemrosesan nilai, diagram juga menunjukkan adanya dua opsi tindakan lain yang dapat diambil pengguna setelah input data, yaitu "Clear" untuk menghapus data yang telah dimasukkan, dan "Keluar" untuk menghentikan atau keluar dari aplikasi tanpa melakukan pemrosesan nilai.

3. Hasil dan pembahasan

Penelitian telah dilakukan dengan hasil implementasi prototipe aplikasi pengolahan nilai mahasiswa yang dikembangkan menggunakan bahasa pemrograman Visual Basic .NET (VB.NET) pada platform Visual Studio. Pembahasan dalam bab ini akan dibagi menjadi dua sub-bab utama. Sub-bab pertama, Pemrograman Visual Basic, akan menguraikan secara rinci aspek teknis pengembangan aplikasi, termasuk struktur kode, logika algoritma yang diimplementasikan untuk pengolahan nilai (perhitungan nilai akhir dan penentuan grade), serta penjelasan mengenai penanganan input dan output data menggunakan komponen-komponen Visual Basic. Sub-bab kedua, Tampilan Program, akan memaparkan desain antarmuka pengguna (user interface) dari prototipe aplikasi. Penjelasan akan mencakup tata letak elemen-elemen antarmuka seperti label, kotak teks, tombol, dan DataGridView, serta pertimbangan desain yang mendasari kemudahan penggunaan dan efisiensi interaksi pengguna dengan aplikasi. Melalui pemaparan pada kedua sub-bab ini, diharapkan dapat memberikan pemahaman yang komprehensif mengenai implementasi teknis dan visual dari prototipe aplikasi pengolahan nilai mahasiswa yang telah dikembangkan.

3.1. Pemrograman Visual Basic

Tampilan pemrograman visual basic disajikan dalam gambar 2 hingga gambar 5. Dimana terdapat 5 bagian utama pada pemrograman visual basic yang telah dilakukan. Bagian pertama adalah coding pemrograman pada bagian awal program, yang ditunjukkan pada gambar 2. Gambar 3 menunjukkan tampilan coding tombol clear, yang akan digunakan untuk reset chatbox kembali ke isian kosong. Gambar 4 merupakan coding untuk tampilan exit, dan gambar 5 merupakan proses pada penghitungan nilai dan penentuan grade nilai.

```
Public Class Form1
    Dim absen, uts, uas As Integer
    Dim nilai, grade As Integer
    0 references
    Private Sub Label1_Click(sender As Object, e As EventArgs) Handles Label1.Click
    End Sub

    0 references
    Private Sub TextBox4_TextChanged(sender As Object, e As EventArgs) Handles lblnilai.TextChanged
    End Sub
```

Gambar 2. Tampilan Awal

```

0 references
Private Sub clear_Click(sender As Object, e As EventArgs) Handles clear.Click
    txtabsen.Clear()
    txtuts.Clear()
    txtuas.Clear()
    lblnilai.Clear()
    lblgrade.Clear()
End Sub

```

Gambar 3. Tampilan Tombol Clear

```

0 references
Private Sub keluar_Click(sender As Object, e As EventArgs) Handles keluar.Click
    Application.Exit()
End Sub

```

Gambar 4. Tampilan Tombol Exit

```

0 references
Private Sub proses_Click(sender As Object, e As EventArgs) Handles proses.Click
    absen = Val(txtabsen.Text * 20 / 100)
    uts = Val(txtuts.Text * 30 / 100)
    uas = Val(txtuas.Text * 50 / 100)
    nilai = absen + uts + uas

    lblnilai.Text = nilai
    lblgrade.Text = grade
    If nilai >= 100 Then
        grade = "A"
    ElseIf nilai >= 90 Then
        lblgrade.Text = "AB"
    ElseIf nilai >= 80 Then
        lblgrade.Text = "B"
    ElseIf nilai >= 75 Then
        lblgrade.Text = "BC"
    ElseIf nilai >= 70 Then
        lblgrade.Text = "C"
    ElseIf nilai >= 65 Then
        lblgrade.Text = "CD"
    ElseIf nilai <= 60 Then
        lblgrade.Text = "D"
    End If
End Sub

```

Gambar 5. Tampilan Format Nilai

3.2. Tampilan Program

Gambar 6 menunjukkan tampilan aplikasi program yang telah di buat. udul "DATA NILAI MAHASISWA" terletak di bagian atas jendela, mengidentifikasi fungsi utama aplikasi. Antarmuka dibagi menjadi dua bagian utama: Input (sebelah kiri) dan Hasil (sebelah kanan). Bagian Input menyediakan kolom-kolom untuk memasukkan data nilai mahasiswa, yang terdiri dari:

- Tugas + Absen: Kotak teks untuk menginput nilai gabungan tugas dan kehadiran mahasiswa.
- UTS: Kotak teks untuk menginput nilai Ujian Tengah Semester.
- UAS: Kotak teks untuk menginput nilai Ujian Akhir Semester.

Bagian Hasil menampilkan output dari perhitungan nilai, yang terdiri dari:

- Nilai Akhir: Kotak teks yang menampilkan hasil perhitungan nilai akhir mahasiswa berdasarkan bobot nilai input.
- Grade: Kotak teks yang menampilkan nilai huruf (grade) yang sesuai dengan nilai akhir (contoh: "D").

Di bagian bawah antarmuka, terdapat tiga Tombol Operasi Aplikasi:

- Proses: Tombol untuk memulai perhitungan nilai akhir dan penentuan grade berdasarkan data input.
- Clear: Tombol untuk menghapus seluruh data input dan output, mengembalikan form ke kondisi awal.
- Keluar: Tombol untuk menutup aplikasi.

Tata letak antarmuka ini dirancang untuk kemudahan penggunaan, dengan pengelompokan input dan output yang jelas serta tombol operasi yang mudah diakses.

Gambar 6. Tampilan Aplikasi

4. Kesimpulan

Berdasarkan pengembangan program aplikasi pengolahan nilai mahasiswa ini, dapat ditarik beberapa kesimpulan utama. Pertama, aplikasi ini berhasil diimplementasikan untuk memfasilitasi pengolahan nilai mahasiswa secara cepat, tepat, dan efisien berdasarkan data Tugas dan Absensi, UTS, serta UAS. Kedua, Visual Basic .NET (VB.NET) dan platform Visual Studio terbukti efektif dalam pengembangan aplikasi berbasis GUI Windows Forms untuk kebutuhan ini. Ketiga, implementasi aplikasi ini berpotensi signifikan dalam mempermudah tugas dosen atau operator dalam menghitung nilai akhir secara otomatis, sekaligus meminimalisir potensi kesalahan perhitungan manual. Terakhir, desain antarmuka pengguna (form) yang sederhana, rapi, dan intuitif berkontribusi pada kemudahan penggunaan aplikasi, dengan pengelompokan input dan output yang jelas untuk meningkatkan efektivitas.

Daftar Pustaka

- [1] Sri Lanka Institute of Information Technology, Malabe, Sri Lanka *et al.*, 'Systematic Review: The Role of Data Analytics in Enhancing Academic Performance Classroom interaction, Learning Analytics in Higher Education', presented at the Proceedings of the 3rd International Conference on Sustainable & Digital Business, Dec. 2024, pp. 168–179. doi: 10.54389/LCDN1827.
- [2] A. Haleem, M. Javaid, M. A. Qadri, and R. Suman, 'Understanding the role of digital technologies in education: A review', *Sustainable Operations and Computers*, vol. 3, pp. 275–285, 2022, doi: 10.1016/j.susoc.2022.05.004.
- [3] J. Hindradjat, I. Hambali, A. Atmoko, and P. H. Livana, 'The Effect of Achievement Motivation and Self-Regulated Learning on Student Learning Behavior in the Time of Online Learning in Indonesia', *Open Access Maced J Med Sci*, vol. 10, no. A, pp. 262–272, Jan. 2022, doi: 10.3889/oamjms.2022.8279.
- [4] T. Gkrimpizi, V. Peristeras, and I. Magnisalis, 'Classification of Barriers to Digital Transformation in Higher Education Institutions: Systematic Literature Review', *Education Sciences*, vol. 13, no. 7, p. 746, Jul. 2023, doi: 10.3390/educsci13070746.
- [5] R. D. S. Damanik, S. N. Kesuma, and M. H. W. Harahap, 'Analisis Sistem Informasi Dalam Mengintegrasikan Peta Wisata ke Dalam Aplikasi Visual Basic.NET', *JRSIKOM*, vol. 1, no. 1, pp. 115–119, Jan. 2025.
- [6] D. Gusrion, S. Kom, and M. Kom, 'MEMBUAT APLIKASI PENYIMPANAN DAN PENGOLAHAN DATA DENGAN VB.NET', vol. 5, no. 1, 2018.
- [7] Politeknik Negeri Bali, K. A. T. Indah, P. G. Sukarata, and Politeknik Negeri Bali, 'Penjadwalan Perkuliahan dengan Metode Metaheuristic Ant Colony Optimization Studi Kasus Politeknik Negeri Bali', *MATRIX*, vol. 9, no. 2, pp. 74–82, Jul. 2019, doi: 10.31940/matrix.v9i2.1335.
- [8] A. M. Najibufahmi and D. M. Prabowoputra, 'Rancang bangun aplikasi kalkulator konversi suhu dengan bahasa visual basic sebagai media bantu dalam pembelajaran', vol. 5, 2025.
- [9] N. S. Yanofsky, 'Towards a Definition of an Algorithm', Jun. 10, 2010, *arXiv: arXiv:math/0602053*. doi: 10.48550/arXiv.math/0602053.
- [10] O. Ferstl, 'Flowcharting by stepwise refinement', *SIGPLAN Not.*, vol. 13, no. 1, pp. 34–42, Jan. 1978, doi: 10.1145/953428.953432.