LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN

PENGENALAN WINDOW BUILDER, GUI, DAN PENGGUNAAN TRY CATCH PADA JAVA

Oleh: Afif Naufal Zahran

NIM: 2511533009

DOSEN PENGAMPU: DR.WAHYUDI, S.T, M.T

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

Kata Pengantar

Pedoman ini disusun sebagai rujukan resmi bagi mahasiswa Departemen Informatika dalam penyusunan laporan praktikum pada mata kuliah *Pemrograman Dasar dengan Java*. Dokumen ini tidak hanya memberikan gambaran umum mengenai format penulisan, tetapi juga menguraikan secara rinci sistematika laporan, tata cara penyajian isi, serta contoh penulisan kode program yang dilengkapi dengan referensi ilmiah. Melalui panduan ini, mahasiswa diharapkan mampu menyusun laporan yang tidak sekadar memenuhi aspek administratif, tetapi juga mencerminkan ketelitian, keteraturan, dan penerapan kaidah penulisan akademik pada tingkat dasar. Dengan demikian, laporan praktikum yang dihasilkan dapat berfungsi sebagai media pembelajaran, dokumentasi kegiatan, sekaligus sarana untuk melatih keterampilan menulis ilmiah yang akan bermanfaat dalam jenjang studi selanjutnya.

Padang, 2025

Tim Penyusun

DAFTAR ISI

BAB I

PENDAHULUAN

1.1 Latar Belakang

Graphical User Interface atau yang sering dikenal dengansingkatan GUI merupakan antarmuka pengguna grafis yang menggunakan elemen visual seperti ikon, menu, dan tombol untuk memungkinkan pengguna berinteraksi dengan perangkat lunak atau system secara intuitif, tanpa perlu mengetikkan perintah teks seperti pada system *Command Line Interface*(CLI). Dengan GUI kita dapat menyederhanakan interaksi, meningkatkan efisiensi, dan membuat penggunaan teknologi lebih mudah dipelajari dan nyaman. Pada banyak bahasa pemograman, kita dapat membuat program yang memiliki GUI salah satunya adalah java, dengan menggunakan bantuan *plugin* pada *eclipse* yakni *WindoBuilder*.

Ketika pengguna memberikan *input* dengan tipe data yang salah, java akan mengeluarkan *output* berupa error yang telah di ekspetasikan. Error tersebut dapat kita ubah menjadi pesan atau justru menjalankan program yang lain ketika error tersebut muncul dengan menggunakan kata kunci *try* dan *catch* pada java

1.2 Tujuan

Tujuan dari laporan praktikum ini agar pembaca dan penulis dapat mendalami dan memahami tentang *WindoBuilder*, GUI, dan penggunaan *try and catch* pada bahasa pemograman *Java*.

1.3 Manfaat

Manfaat dari praktikum tentang *WindowBuilder* dan penggunaan *try catch* agar pembaca mendapatkan pengetahuan tentang *penggunaan WindowBuilder* dan *try catch*. Manfaat untuk penulis agar dapat memahami penggunaan *WindowBuilder* dan *try catch* pada bahasa pemograman *Java*.

BAB II
PEMBAHASAN

2,1 Teori

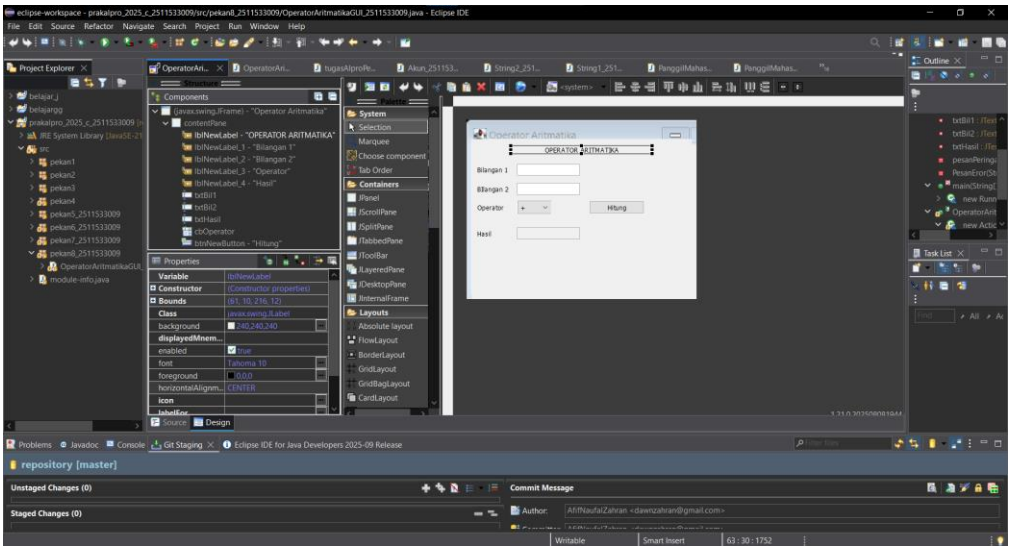
Window Builder adalah suatu *plugin* yang memudahkan pengguna dalam membangun suatu program yang menggunakan GUI(*Graphical Interface Interface*) dengan menyediakan visualisasi design dan juga *code generation*. Tidak hanya itu, kita dapat mengubah kode yang telah tergenerasi sehingga memberikan fleksibilitas dalam pembuatan program GUI.

Pada java juga terdapat mekanisme dasar yang berfungsi untuk menangani pengecualian atau error yakni *try* dan *catch* dimana ketika terjadi error kita dapat melakukan pengecualian dan menjalankan program untuk menangani error atau masalah tersebut.

2,2 Program

Berikut penggunaan WindowBuilder untuk membuat program GUI menghitung dua bilangan dengan *input* bilangan dan operator dari pengguna dengan nama file OperatorAritmatikaGUI_2511533009.

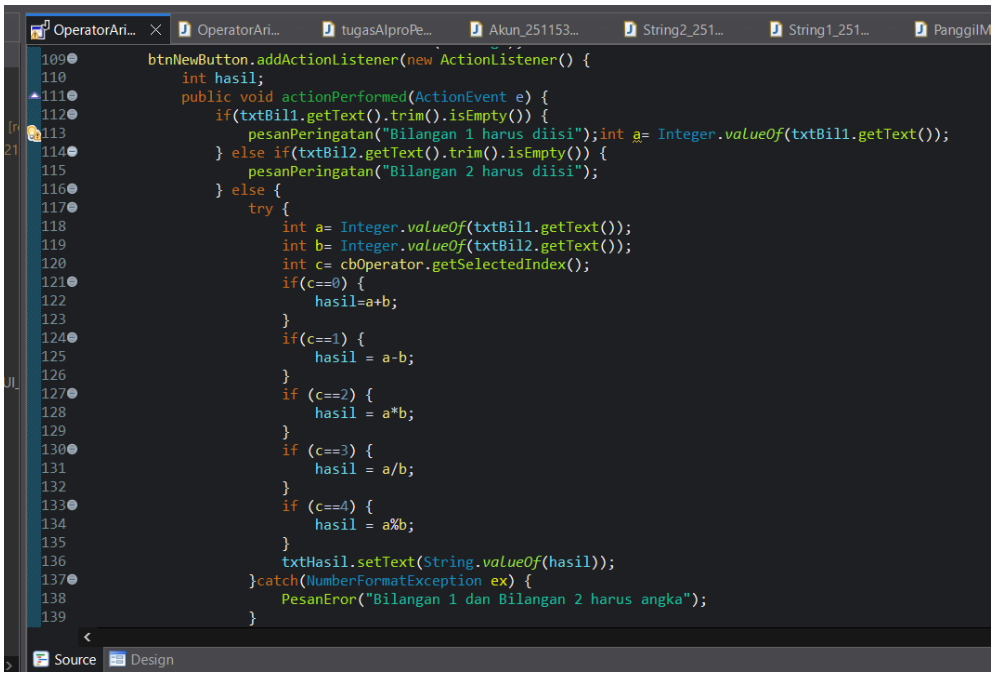
a. Design



Gambar 2.1

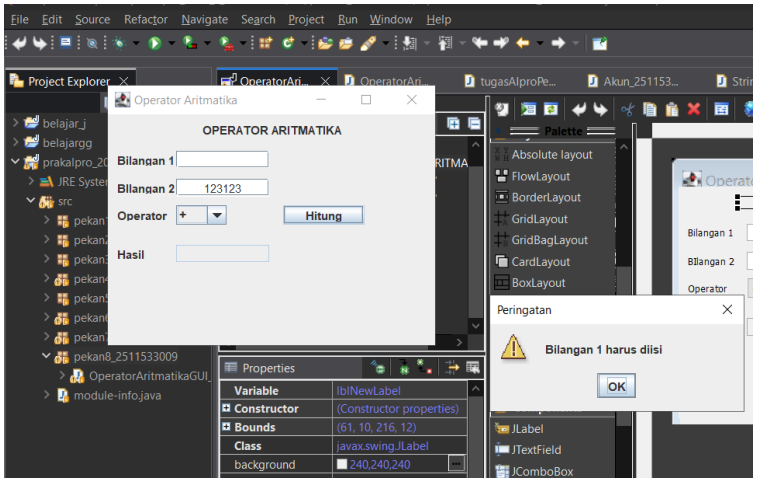
Pada gambar 2.1 merupakan tampilan design dari *WindowBuilder* dimana tedapat bagian *Component* befungsi untuk menampilkan komponen yang digunakan, area *Properties* untuk menampilkan properti dari komponen yang kita gunakan, area untuk menambahkan komponen apa yang akan kita gunakan, dan area tampilan untuk mevisualisasikan GUI yang hendak kita buat. Terdapat 4 komponen yang digunakan yakni *JLabel* untuk menampilkan teks, *JTextField* untuk menerima *input* dan mengeluarkan *output*, *JComboBox* untuk memilih operator apa yang akan digunakan, dan *JButton* sebagai tombol.

b. Source

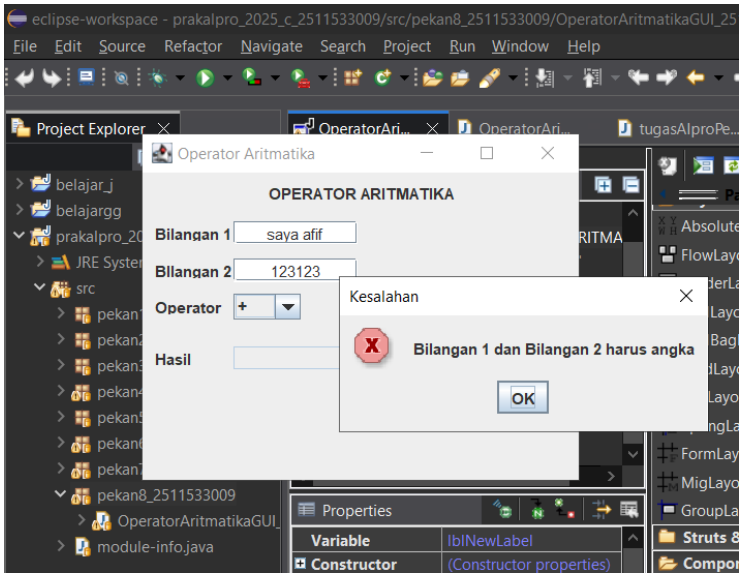


Gambar 2.2

Pada gambar 2.2 memanggil fungsi dari variabel *btnNewButton* dimana ketika pengguna menekan tombol “Hitung” program akan menjalankan program yang kita buat. Sebelum program melakukan operasi aritmatika, proram akan melakukan pengecekan apakah *input* kosong jika ya maka akan muncul GUI peringatan dengan pesan “Bilangan [angka] harus diisi” jika tidak maka kita akan menggunakan *try* dan *catch* untuk melakukan pengecekan apakah nilai *input* berupa angka atau terdapat *character* selain angka, jika ya maka program akan lanjut mengeksekusi kode operasi aritmatika, jika tidak program akan menampilkan GUI pesan eror dengan memanggil *helper method* “PesanEror”.



Gambar 2.3



Gambar 2.4

BAB III

PENUTUPAN

3,1 Kesimpulan

WindowBuilder sangat berguna karena memudahkan pengguna untuk membangun suatu aplikasi GUI. *Try* dan *catch* pada bahasa pemrograman *Java* sangat penting karena merupakan hal yang sangat fundamental pada bahasa pemrograman terutama bahasa pemrograman *Java*.

3,2 Saran

Laporan praktikum ini masih memiliki kekurangan. Karena itu, penulis sangat terbuka terhadap saran dan kritikan agar dapat meningkatkan kualiats laporan ini dan laporan-laporan selanjutnya.