



STUDENT EMAIL

GUI Linked List

Bilal
1910130007

Akbar
1910130011

Zulfikri
1810130008

PENDAHULUAN

Email atau surat elektronik adalah suatu sarana untuk mengirim dan menerima surat atau pesan dengan format digital melalui jalur jaringan komputer dan internet. Pesan digital atau surat elektronik tersebut dapat dibuat dan dikirim ke alamat email lainnya dengan menggunakan komputer/ laptop dan gadget lainnya. Email menjadi tempat pengiriman pesan terfavorit karena selain pengirimannya yang cepat, user tidak perlu mengeluarkan banyak biaya untuk ongkos kirim. Namun, karena cepat dan gratisnya pengiriman, membuat seorang user terkadang menerima banyak Email yang bahkan bisa menjadi spam atau mengganggu kenyamanan user itu sendiri. Di era digital yang massive ini, diperlukan suatu perintah untuk menlist atau memfilter surat-surat yang dianggap sebagai spam atau tidak. Maka dari itu, kami akan menggunakan single Linked list untuk menambah, mengurangi, dan menampilkan email-email. Single linked list adalah bentuk struktur data yang berisi kumpulan data yang biasa disebut sebagai node yang saling sambung menyambung. Kami akan menggunakan Linked list ini sebagai tugas akhir kami.

ABSTRAK

Email menjadi salah satu fitur terpenting dalam era modern ini. Sebagai pengganti surat pos, Email menjadi fitur yang unik dan disukai oleh user karena proses pengirimannya yang cepat dan gratis. Namun, di era modern saat ini Email juga berisikan oleh surat-surat yang bisa dibilang spam yang memenuhi kotak masuk Email. Maka dari itu, diperlukannya filter ataupun perintah yang bisa menlist email-email yang massive sehingga user tidak perlu terganggu oleh surat-surat yang spam. Dengan menggunakan GUI Linked list, kami akan membuat Linked list sebagai sarana untuk menlist email yang massive.

TEORI DASAR

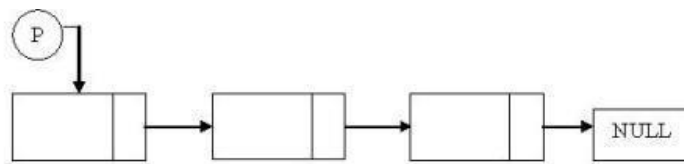
Linked List

a. Nodes

Self-referential objects (object yang mereferensikan dirinya sendiri) yang disebut nodes, yang dihubungkan dengan links, membentuk kata “linked” list.

b. Linked List (LL)

Adalah koleksi data item yang tersusun dalam sebuah barisan secara linear, dengan penyisipan dan pemindahan dapat dilakukan dalam semua tempat di LL tersebut.



Single Linked List

Adalah sebuah LL yang menggunakan sebuah variabel pointer saja untuk menyimpan banyak data dengan metode LL, suatu daftar isi yang saling berhubungan.

Ilustrasi single LL:

Pada gambar di atas, data terletak pada sebuah lokasi dalam sebuah memory, tempat yang disediakan memory untuk menyimpan data disebut node ? simpul, setiap node memiliki pointer (penunjuk) yang menunjuk ke node berikutnya sehingga terbentuk suatu untaian yang disebut single LL.

Bila dalam single LL pointer hanya dapat bergerak ke satu arah saja, maju / mundur, kanan / kiri, sehingga pencarian datanya juga hanya satu arah saja.

Graphic User Interface

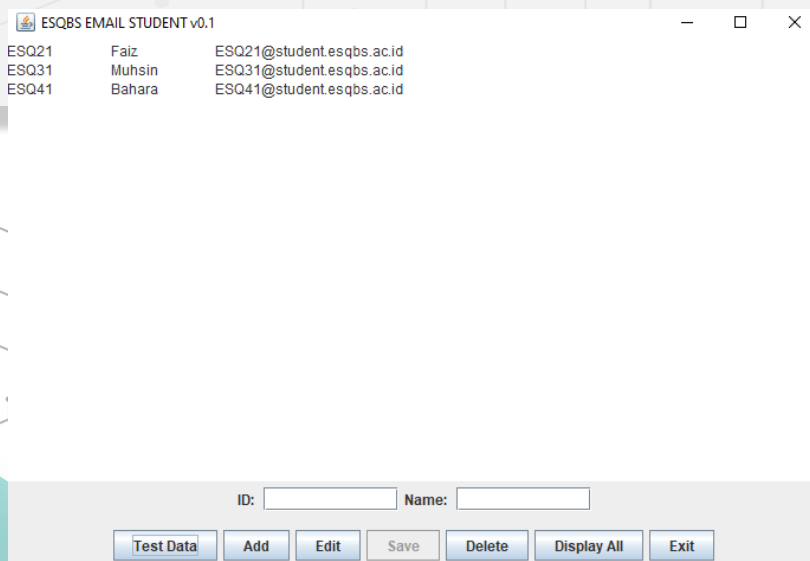
Pengertian GUI (Graphical User Interface) adalah bentuk antarmuka pengguna untuk memungkinkan user dapat berinteraksi dengan perangkat elektronik.

GUI memiliki beberapa elemen, mulai dari elemen windows, menu, icon, widget dan juga tab. Untuk menggunakan elemen ini biasanya GUI akan mendapatkan inputan dari perangkat masukan, baik secara manual maupun dengan teknologi touchscreen.

Dalam perkembangannya, GUI akan terus dikembangkan untuk semakin memudahkan penggunaannya seperti misalnya pengembangan teknologi gesture dan juga teknologi remote jarak jauh sebagai salah satu cara inputan di masa depan.

GUI merupakan salah satu jenis user interface yang digunakan untuk melakukan komunikasi antara manusia dengan perangkat seperti laptop, komputer, ponsel dan tablet. Hal ini menjadikan komponen GUI selalu berhubungan dengan representasi visual dari sebuah sistem operasi ataupun software.

GUI sangat berperan penting serta sangat memudahkan user untuk mengoperasikan sebuah sistem atau software aplikasi dengan navigasi yang ada. Untuk itu penting bagi Anda mengetahui lebih detail tentang apa itu pengertian, fungsi, jenis, komponen, kelebihan dan kekurangan dari graphical user interface.



Graphic User Interface

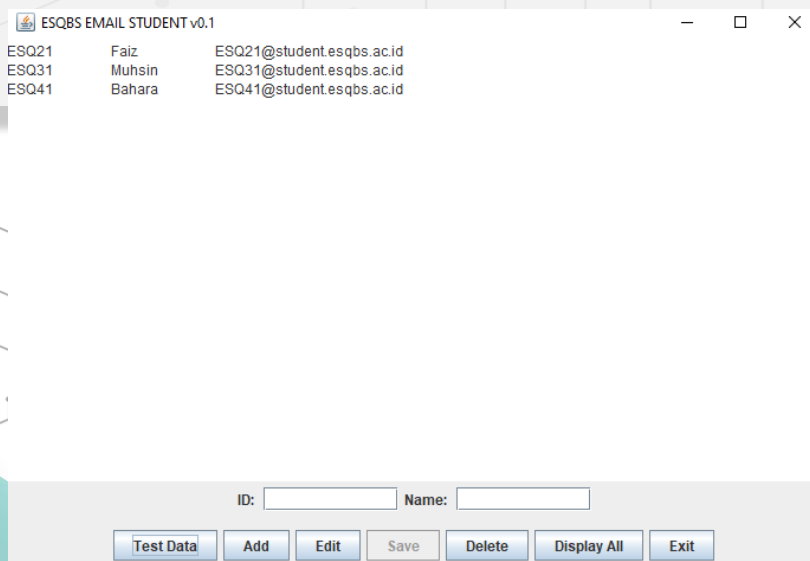
Pengertian GUI (Graphical User Interface) adalah bentuk antarmuka pengguna untuk memungkinkan user dapat berinteraksi dengan perangkat elektronik.

GUI memiliki beberapa elemen, mulai dari elemen windows, menu, icon, widget dan juga tab. Untuk menggunakan elemen ini biasanya GUI akan mendapatkan inputan dari perangkat masukan, baik secara manual maupun dengan teknologi touchscreen.

Dalam perkembangannya, GUI akan terus dikembangkan untuk semakin memudahkan penggunaannya seperti misalnya pengembangan teknologi gesture dan juga teknologi remote jarak jauh sebagai salah satu cara inputan di masa depan.


GUI merupakan salah satu jenis user interface yang digunakan untuk melakukan komunikasi antara manusia dengan perangkat seperti laptop, komputer, ponsel dan tablet. Hal ini menjadikan komponen GUI selalu berhubungan dengan representasi visual dari sebuah sistem operasi ataupun software.

GUI sangat berperan penting serta sangat memudahkan user untuk mengoperasikan sebuah sistem atau software aplikasi dengan navigasi yang ada. Untuk itu penting bagi Anda mengetahui lebih detail tentang apa itu pengertian, fungsi, jenis, komponen, kelebihan dan kekurangan dari graphical user interface.



PEMBAHASAN

Fitur

 ESQBS EMAIL STUDENT v0.1

ESQ21	Faiz	ESQ21@student.esqbs.ac.id
ESQ31	Muhsin	ESQ31@student.esqbs.ac.id
ESQ41	Bahara	ESQ41@student.esqbs.ac.id

ID:

Name:

Test Data

Add

Edit

Save

Delete

Display All

Exit

Test data Add Edit Delete Display All

PEMBAHASAN

Gui Component

```
public class StudentEmailGUI extends JFrame
{
    // Constants:

    // GUI Componet:
    JTextArea studentTextArea = new JTextArea ();

    JLabel idLabel = new JLabel ("ID: ");
    JTextField idTextField = new JTextField (10);
    JLabel nameLabel = new JLabel ("Name: ");
    JTextField nameTextField = new JTextField (10);

    JButton testDataButton = new JButton ("Test Data");
    JButton addButton = new JButton ("Add");
    JButton deleteButton = new JButton ("Delete");
    JButton editButton = new JButton ("Edit");
    JButton editSaveButton = new JButton ("Save");
    JButton displayAllButton = new JButton ("Display All");
    JButton exitButton = new JButton ("Exit");
}
```

Pada project ini GUI yang kami pakai kurang lebih seperti gambar di atas , kita memakai JLabel , JTextarea, JTextField, JButton.

Gui pada dasarnya sangat memudahkan pekerjaan kita dan menjadikan tampilan lebih menarik

PEMBAHASAN

Fitur Test List

```
private void addTestData ()  
{  
    nameTextField.setText ("Faiz");  
    idTextField.setText  ("ESQ21");  
    addStudent ();  
  
    nameTextField.setText ("Muhsin");  
    idTextField.setText  ("ESQ31");  
    addStudent ();  
  
    nameTextField.setText ("Bahara");  
    idTextField.setText  ("ESQ41");  
    addStudent ();  
}
```

Test list berguna untuk menampilkan data yang sudah di sediakan di dalam program , guna untuk melihat atau mentest apakah program ini dapat berjalan dengan baik atau tidak.

PEMBAHASAN

Fitur Add

```
private void addStudent ()
{
    if (isStudentIdInLinkedList (idTextField.getText()) == true)
    {
        JOptionPane.showMessageDialog (StudentEmailGUI.this,
            "Error: student ID is already in the database.");
    }
    else
    {
        try
        {
            StudentEmail stud = new StudentEmail (nameTextField.getText(),
                idTextField.getText());

            studentLinkedList.add (stud);                // tambah student

            displayAll ();

            nameTextField.setText("");
            idTextField.setText("");
        }
        catch (StudentEmailException error)
        {
            JOptionPane.showMessageDialog (StudentEmailGUI.this, error.toString ());
            // myLabel.setText (error.toString ());
        }
    }
}
```

Fitur ini menggunakan Linked list dengan method

studentLinkedList.add (stud);

Fungsinya adalah menambahkan data email mahasiswa ke database yang telah di sediakan, mengambil data (get) dari kelas studentLinkedList lalu mengisi di *class* studentEmail.java

```
StudentEmail stud = new StudentEmail (nameTextField.getText(),
    idTextField.getText());
```

PEMBAHASAN

Fitur Delete

```
private void deleteStudent ()
{
    if (studentLinkedList.size() == 0)
    {
        JOptionPane.showMessageDialog (StudentEmailGUI.this,
        "Error: Database is empty.");
    }
    else if (isStudentIdInLinkedList (idTextField.getText()) == false)
    {
        JOptionPane.showMessageDialog (StudentEmailGUI.this,
        "Error: student ID is not in the database.");
    }
    else
    {
        for (int s = 0; s < studentLinkedList.size(); s++)
        {
            String currId = studentLinkedList.get (s).getId ();

            if (currId.compareToIgnoreCase (idTextField.getText()) == 0)
            {
                studentLinkedList.remove (s);           // menghapus
            }
        }

        displayAll ();

        nameTextField.setText("");
        idTextField.setText("");
    }
}
```

Fitur ini menggunakan Linked list dengan method

`studentLinkedList.remove (s);`

Fungsinya adalah menghapus data email mahasiswa ke database yang telah di sediakan, menghilangkan data (get) dari kelas studentLinkedList lalu menghilangkan di *class* studentEmail.java

PEMBAHASAN

Fitur Edit Save

```
// Class Instance Data:  
private LinkedList<StudentEmail> studentLinkedList = new LinkedList<StudentEmail> ();  
private int editIndex;
```

```
private void editSaveStudent ()  
{  
    // This code will preserve the changes the user made to the student  
    // they were editing - and save them back into the Linked List.  
  
    studentLinkedList.get (editIndex).setName (nameTextField.getText() );  
    studentLinkedList.get (editIndex).setId (idTextField.getText() );  
  
    displayAll ();  
  
    nameTextField.setText ("" );  
    idTextField.setText ("" );  
  
    editSaveButton.setEnabled (false);  
}
```

Fitur ini berfungsi mengedit data yang sudah ada pada daftar linked list , lalu data bisa di save Kembali dengan id maupun nama yang berbeda

PEMBAHASAN

Exception

```
public class StudentEmailException extends Exception
{
    public StudentEmailException ()
    {
        super ();
    }

    public StudentEmailException (String message)
    {
        super (message);
    }

    public StudentEmailException (Throwable cause)
    {
        super (cause);
    }

    public StudentEmailException (String message, Throwable cause)
    {
        super (message, cause);
    }
}
```

Fungsinya menhandle bila terjadinya sebuah eror , missal karena kesalahan dalam mengisi Id ataupun nama. Berikut exception errornya:

1. Ketika Id text field dan nama text field kosong.
2. Ketika nama atau id sudah terdaftar di database.
3. Ketika salah dalam menginput id untuk mengganti value dalam id maupun nama.

PENUTUPAN

Kesimpulan

Kesimpulan yang bisa kami ambil adalah era digital ini sudah banyak email yang perlu di list. melalui Linked list, kami bisa menambah, menghapus, dan menampilkan email yang massive

Saran

Adapun saran yang bisa kami sampaikan adalah diharapkannya agar fitur ini bisa tersedia agar user bisa memilah-milah email yang benar-benar informasi dan email yang dianggap spam.



ありがとう

THANK YOU