

## BAB II

### GRAPHIC USER INTERFACE

#### A. KOMPETENSI DASAR

- 3.1 Memahami pembuatan Graphic User Interface (GUI)
- 4.1 Menyajikan komponen-komponen GUI dengan pengendalian aplikasi dengan pengguna

#### B. TUJUAN PEMBELAJARAN

Setelah mempelajari bab ini, siswa diharapkan mampu:

1. Siswa dapat memahami konsep graphical user interface
2. Siswa dapat memahami hiraku class graphical user interface
3. Siswa dapat memahami komponen dasar GUI
4. Siswa dapat memahami komponen delegation event model
5. Siswa dapat memahami penanganan kejadian ( event handling)
6. Siswa dapat memahami class-class adapter

#### MATERI PEMBELAJARAN

##### 1. API UNTUK APLIKASI GUI

###### AWT (Abstract Window Toolkit):

- a. Library dan komponen GUI (java.awt) yang pertama kali diperkenalkan oleh Java, Sun tidak merekomendasikan lagi penggunaan komponen GUI dari AWT

###### Swing or JFC (Java Foundation Class):

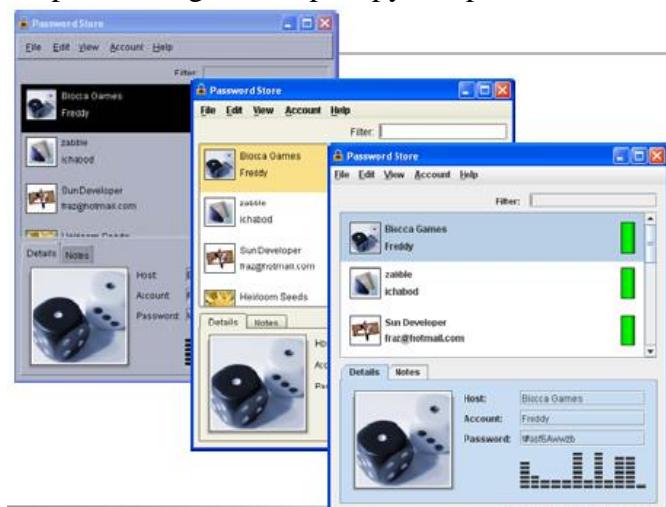
- b. Library dan komponen GUI (javax.swing) terbaru dari Java dan yang direkomendasikan Sun untuk pemrograman GUI. Komponen Swing sebagian besar adalah turunan AWT dan lebih lengkap daripada AWT

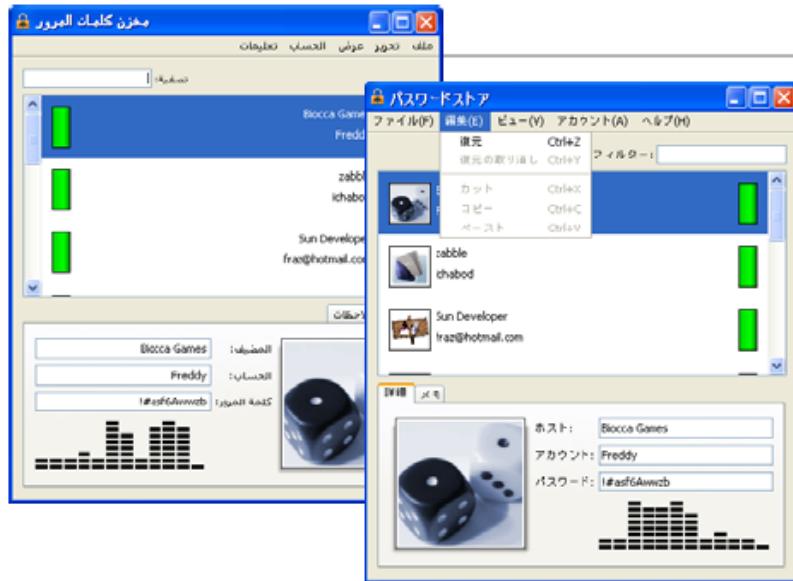
##### 2. FITUR SWING

Komponen GUI Lengkap: button, listbox, combobox, textarea, dsb

Pluggable Look-and-Feel: tampilan GUI dapat diubah sesuai dengan kehendak (tidak perlu mengikuti native OS)

Data Transfer Antar Komponen: drag and drop, copy and paste





### 3. KOMPONEN SWING

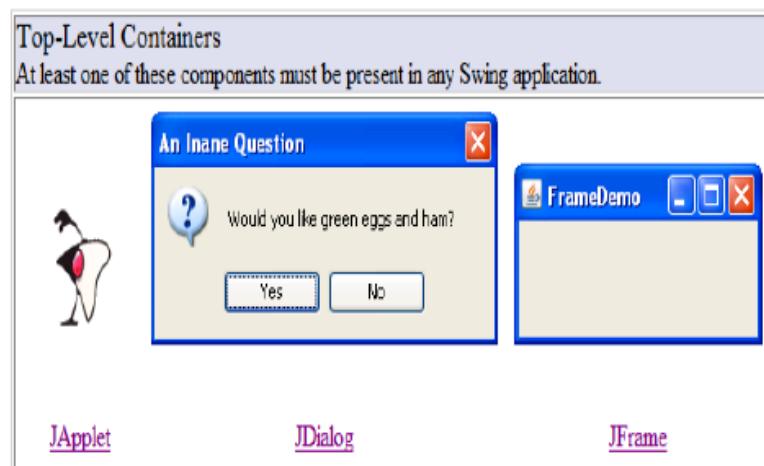
**Top-Level Container:** kontainer dasar dimana komponen lainnya diletakkan (JFrame, JDialog dan Applet)

**Intermediate Container:** kontainer perantara dimana komponen lainnya diletakkan/dikelompokkan (JPanel, JScrollPane, JTabbedPane, JToolbar, JSplitPane)

**Atomic Component:** komponen yang memiliki fungsi spesifik dan menerima interaksi langsung dari user (Jbutton, JLabel, JTextArea, dsb)

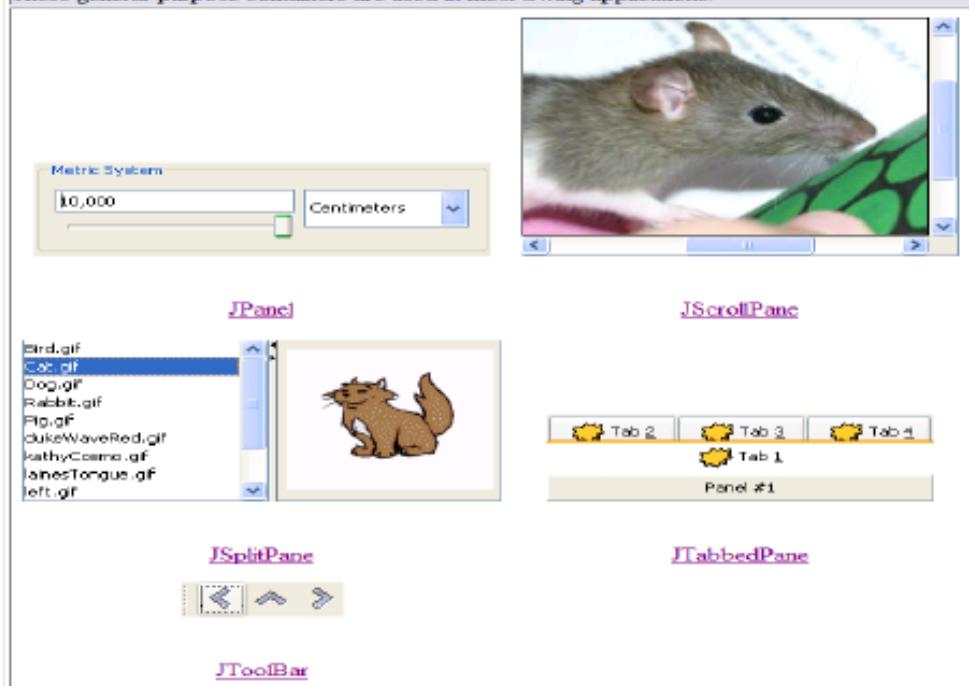
**Layout Manager:** mengatur tata letak dan posisi komponen dalam kontainer (BorderLayout, BoxLayout, FlowLayout, GridBagLayout, GridLayout)

**Event Handling:** menangani event yang dilakukan user (klik mouse, ketik keyboard, perbesar frame, dsb)



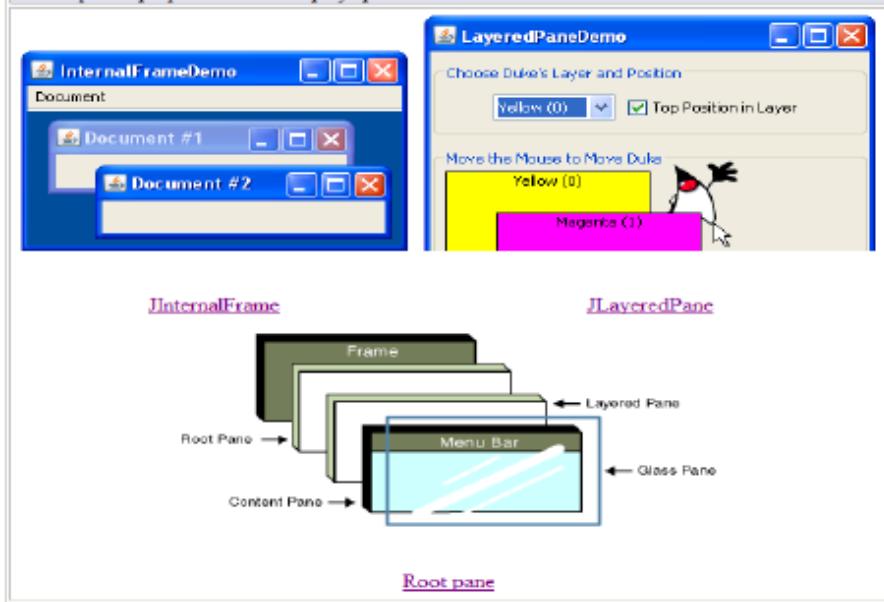
### General-Purpose Containers

These general-purpose containers are used in most Swing applications.



### Special-Purpose Containers

These special-purpose containers play specific roles in the UI.



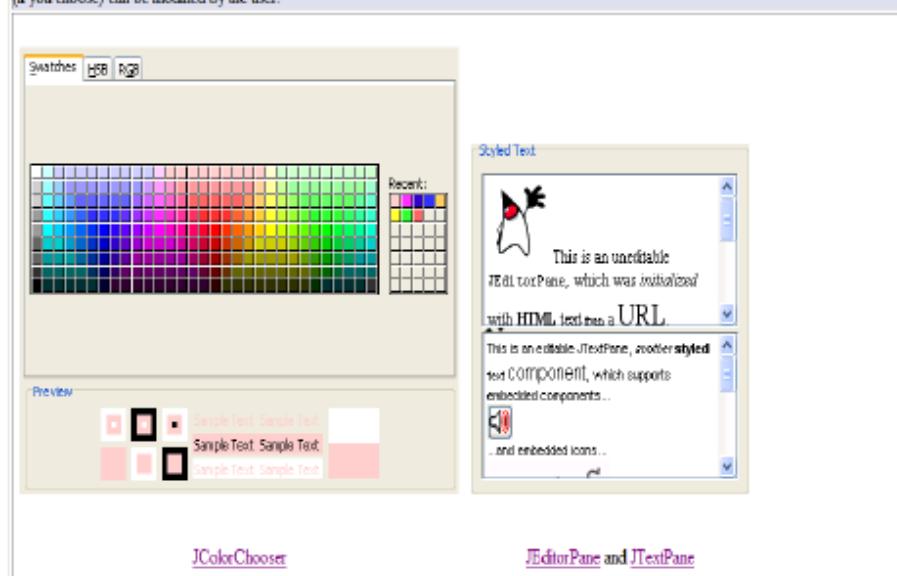
### Basic Controls

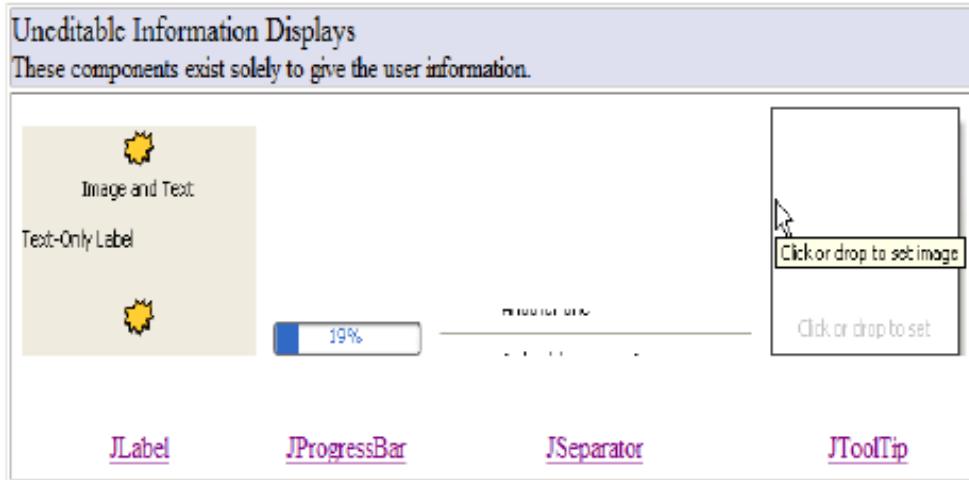
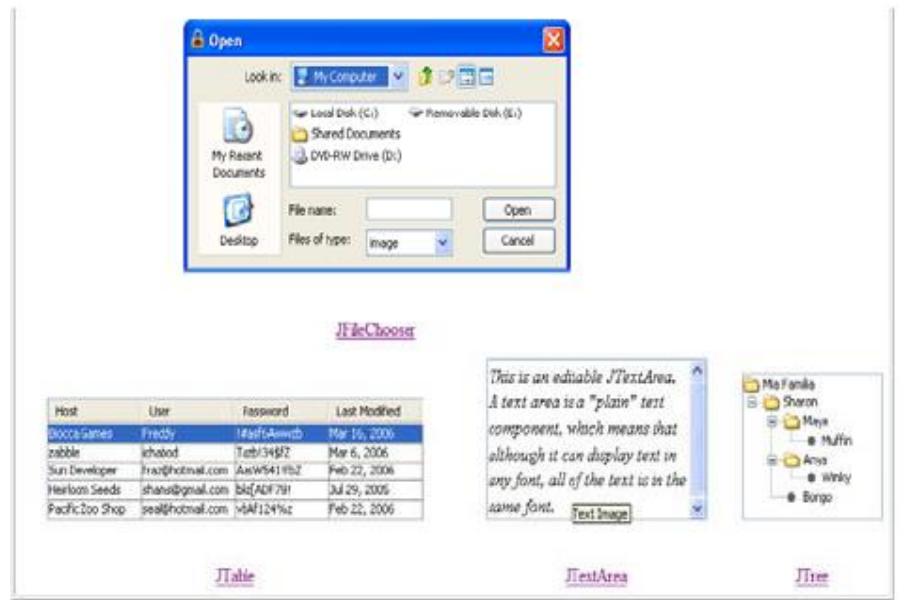
Simple components that are used primarily to get input from the user; they may also show simple state.



### Interactive Displays of Highly Formatted Information

These components display highly formatted information that (if you choose) can be modified by the user.





#### 4. PENGGUNAAN TOP LEVEL CONTAINER

Komponen GUI akan ditampilkan sebagai bagian dari containment hierarchy (Susunan hirarki komponen yang menjadikan top-level container sebagai rootnya)

Setiap komponen GUI hanya dapat dimuat pada satu top-level container. Dan setiap top-level container memiliki satu content pane yang berisi komponen-komponen GUI

Kita dapat menambahkan menubar pada top-level container. Menubar secara konvensi ditempatkan di dalam top-level container, tapi di luar content pane, kecuali pada look and feel MacOS yang memberi kita opsi penempatan lebih bebas

#### 5. CONTAINMENT HIERARCHY

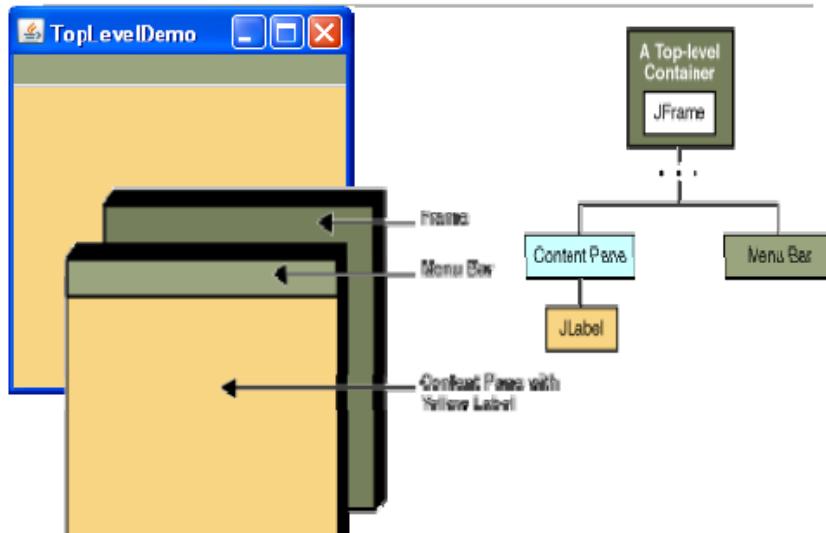
Aplikasi GUI berbasis Swing paling tidak memiliki satu containment hierarchy dengan

JFrame sebagai rootnya

Jika sebuah aplikasi memiliki satu window utama dan dua dialog artinya aplikasi tersebut memiliki tiga containment hierarchy dan juga tiga top-level container

Satu containment hierarchy memiliki JFrame sebagai root, dan dua yang lain memiliki JDialog sebagai root

## 6. TAMPILAN UMUM TOP LEVEL CONTAINER



## 7. CONTENT PANE DAN MENU BAR

Menambahkan content pane pada Jframe frame.getContentPane().add(yellowLabel , BorderLayout.CENTER);

Menambahkan komponen pada content pane

```
JPanel contentPane = new JPanel(new BorderLayout());  
contentPane.setBorder(someBorder );  
contentPane.add(someComponent , BorderLayout.CENTER);  
contentPane.add(anotherComponent , BorderLayout.PAGE_END);  
frame.setContentPane(contentPane);
```

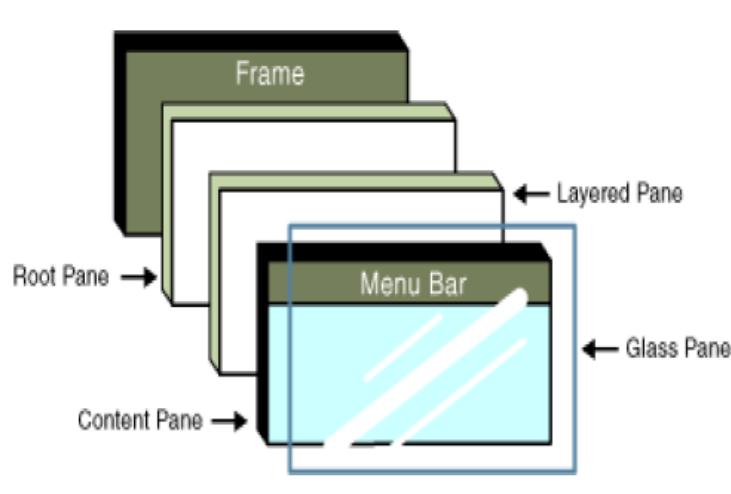
Menambahkan menu bar pada Jframe frame.setJMenuBar(greenMenuBar);

## 8. ROOT PANE

Setiap top-level container sebenarnya didukung oleh satu intermediate container bernama root pane

Root pane mengelola content pane, menu bar dan dua kontainer lain (layered pane dan glass pane)

Layered pane mengatur kedalaman posisi komponen dengan menggunakan nilai depth (int)  
Glass pane sering digunakan untuk menangkap kejadian atau menempatkan gambar pada suatu wilayah dimana terdapat satu atau banyak komponen



## 9. COMPONENT

**Basic Controls**  
Simple components that are used primarily to get input from the user; they may also show simple state.

**JButton**      **JCheckBox**      **JComboBox**      **JList**

**JMenu**      **Another Menu**  
A text-only menu item Alt+1  
Both text and icon  
A radio button menu item Another one  
A check box menu item Another one  
A submenu

**JSlider**

**JSpinner**

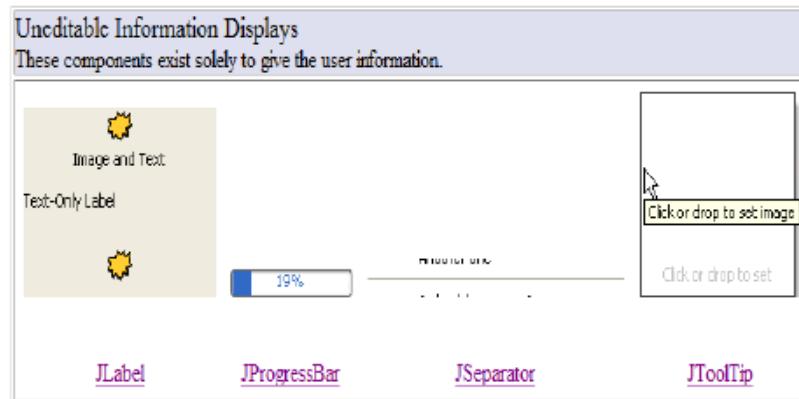
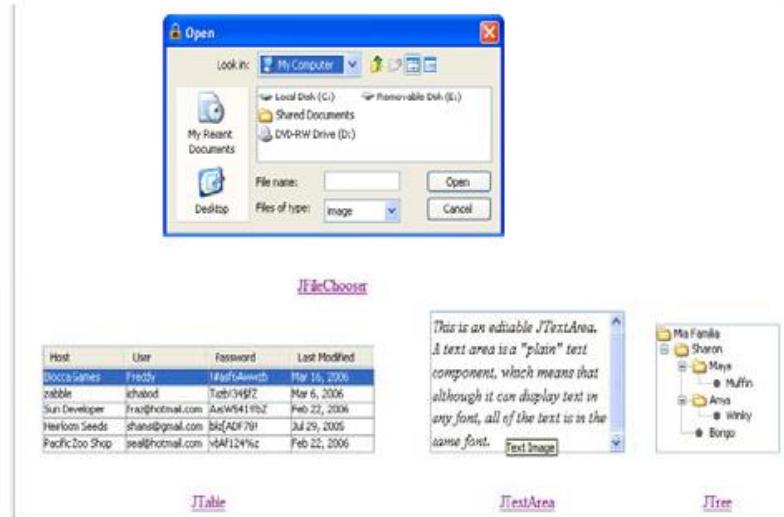
**JTextField**

**JPasswordField**

**Interactive Displays of Highly Formatted Information**  
These components display highly formatted information that (if you choose) can be modified by the user.

**JColorChooser**

**EditorPane and TextPane**



## **TUGAS INDIVIDU**

### **TUGAS 1**

1. Apa yang dimaksud dengan User Interface?
2. Apa yang dimaksud dengan User Friendly?
3. Sebutkan dan jelaskan dua macam user interface!
4. Sebutkan dan jelaskan karakteristik GUI!
5. Sebutkan dan jelaskan tipe-tipe interaksi dengan User!

### **TUGAS 2**

1. Sebutkan dan jelaskan komponen dasar UI dengan Swing!
2. Jelaskan 2 point Delegation Event Model!
3. Sebutkan 3 komponen penting Delegation Event Model!
4. Sebutkan 4 macam Event Listeners!
5. Sebutkan 8 Class Event!

### **TUGAS 3**

Buatlah aplikasi pengolah database (Ms. Access atau My SQL) menggunakan Java Netbeans dengan kriteria minimal 1 form input, 1 form transaksi dan 1 laporan!