**BAB II**

**JAVABEAN & GIT**

**2.1 Tujuan**

1. Praktikan mengetahui fungsi dari java bean.
2. Praktikan mengetahui property dari java bean.
3. Praktikan mengetahui dan memahami tentang apa itu java bean.
4. Praktikan mampu membuat dan memanfaatkan bean pada java bean.
5. Praktikan dapat mengimplementasikan komponen dari java bean ke dalam sebuah aplikasi.
6. Praktikan dapat mengetahui penggunaan command Git dan Github

**2.2 Alat dan Bahan**

**2.2.1 Laptop**



Gambar 2.1 laptop atau pc

Digunakan sebagai media untuk membuat project pada aplikasi netbeans

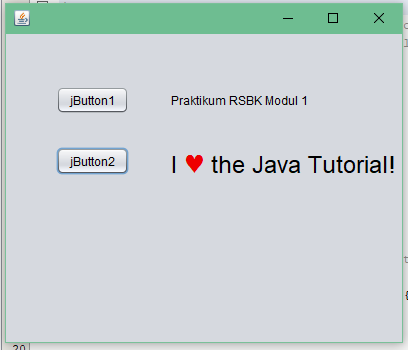
**2.2.2 NetBeans**



Gambar 2.2 NetBeans

Aplikasi yang digunakan untuk menulis, meng-compile, mencari kesalahan dan menyebarkan program netbeans yang ditulis dalam Bahasa pemograman java.

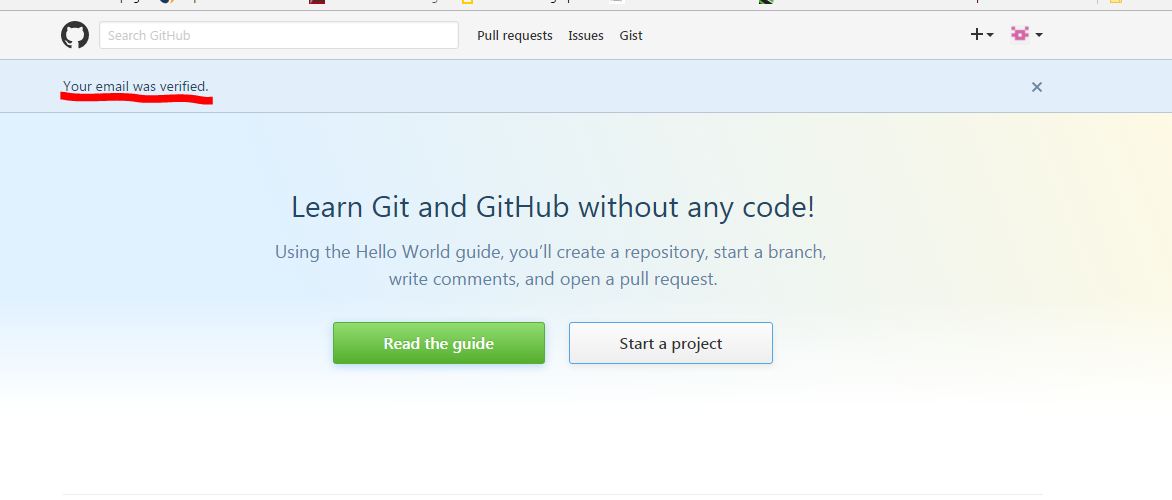
**2.2.3 BumperSticker.Jar**



Gambar 2.3 BumperStiker.Jar

BumperSticker adalah sebuah label (stiker) yang berisi pesan yang berlangsung dapat digunakan ketika ditambahkan pada pallete dalam Netbeans.

**2.2.4 Akun Github yang sudah terverifikasi**



Gambar 2.4 Akun Github

Github adalah  software Hosting untuk proyek open source  yang menggunakan Tool  System revisi kontrol Git.  Jadi Git adalah tool untuk melakukan revisi code

**2.2.5 Git**



Gambar 2.5 git

Git adalah version control system yang digunakan para developer untuk mengembangkan software secara bersama-bersama. Fungsi utama git yaitu mengatur versi dari source code program anda dengan mengasih tanda baris dan code mana yang ditambah atau diganti.

**2.3 Dasar Teori**

**2.3.1 Java Bean**

JavaBean adalah komponen dalam java yang bersifat reusable. JavaBean digunakan untuk mengelola data flow antara aplikasi client atau applet dan komponen yang berjalan pada J2EE server atau antara komponen server dan database. Komponen JavaBean memiliki variabel instan serta method get dan set untuk mengakses data yang disimpan dalam variabel instan. Komponen JavaBean memiliki rancangan dan implementasi yang sederhana.

Pada Dasarnya JavaBeans dapat dibuat dengan cara yang sama dengan pembuatan class Java standar, tetapi kelas Java Beans memiliki beberapa kententuan berikut ini:

1. Class tersebut harus memiliki satu buah konstruktor yang tidak memakai argument.
2. Setiap property dari JavaBeans yang akan diakses oleh public harus memiliki metode khusus untuk menentukan nilainya (metode setter), dan metode khusus untuk mengambil nilainya (metode getter).

Kalau dilihat dari penjelasan di atas, JavaBeans adalah sebuah cara untuk membuatkan objek dari kelas yang dimaksud. Namun objek ini tidak langsung “frontal” dituliskan dalam halaman JSP-nya. Frontal di sini adalah dengan menggunakan sintaks “new” pada halaman JSP-nya, tetapi ada sintaks khusus untuk menggunakan sebuah instansiasi Objek JavaBeans-nya.

Sumber : *http://fahmy-inspiration.blogspot.com/2013/02/java-beans\_28.html*

**2.3.2 Swing/AWT**

Abstract Windowing Toolkit (AWT), atau disebut juga “Another Windowing Toolkit”, adalah pustaka windowing bertujuan umum dan multiplatform serta menyediakan sejumlah kelas untuk membuat GUI di Java. Dengan AWT, dapat membuat window, menggambar, manipulasi gambar, dan komponen seperti Button, Scrollbar, Checkbox, TextField, dan menu pull-down. Swing merupakan perbaikan kelemahan di AWT. Banyak kelas swing menyediakan komponen alternatif terhadap AWT. Contohnya kelas JButton swing menyediakan fungsionalitas lebih banyak dibanding kelas Button. Selain itu komponen swing umumnya diawali dengan huruf “J”, misalnya JButton, JTextField, JFrame, JLabel, JTextArea, JPanel, dan sebagainya. Teknologi swing menggunakan dan memperluas gagasan-gagasan AWT. Sementara, penggunaan komponen Swing ditandai dengan adanya instruksi : import javax.swing.

Sumber : *http://renamuslimahmihardjo.blogspot.com/2012/10/belajar-awt-dan-swing.html*

**2.3.3 GIT**

Git adalah version control system yang digunakan para developer untuk mengembangkan software secara bersama-bersama. Fungsi utama git yaitu mengatur versi dari source code program anda dengan mengasih tanda baris dan code mana yang ditambah atau diganti. Git ini sebenernya memudahkan programmer untuk mengetahui perubahan source codenya daripada harus membuat file baru seperti Program.java, ProgramRevisi.java,  ProgramRevisi2.java, ProgramFix.java. Selain itu, dengan git kita tak perlu khawatir code yang kita kerjakan bentrok, karena setiap developer bias membuat branch sebagai workspacenya.Fitur yang tak kalah keren lagi, pada git kita bisa memberi komentar pada source code yang telah ditambah/diubah, hal ini mempermudah developer lain untuk tahu  kendala apa yang dialami developer lain.

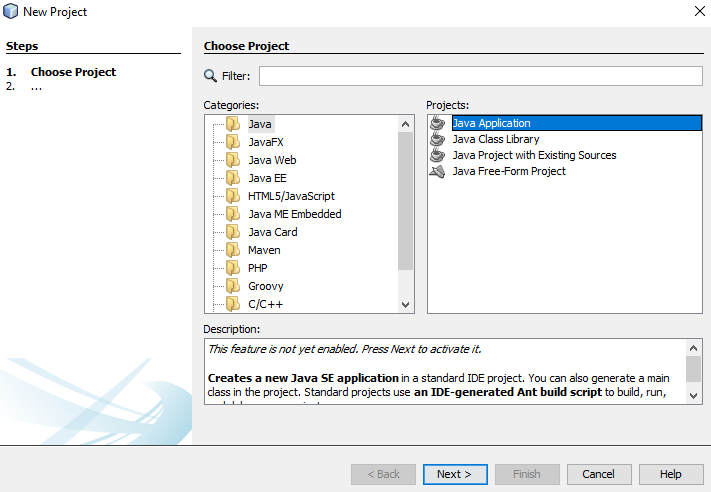
Untuk mengetahui bagaimana menggunakan git, berikut perintah-perintah dasar git:

* Git init : untuk membuat repository pada file lokal yang nantinya ada folder .git
* Git status : untuk mengetahui status dari repository lokal
* Git add : menambahkan file baru pada repository yang dipilih
* Git commit : untuk menyimpan perubahan yang dilakukan, tetapi tidak ada perubahan pada remote repository.
* Git push : untuk mengirimkan perubahan file setelah di commit ke remote repository.
* Git branch : melihat seluruh branch yang ada pada repository
* Git checkout : menukar branch yang aktif dengan branchyang dipilih
* GIt merge : untuk menggabungkan branch yang aktif dan branch yang dipilih
* Git clone : membuat Salinan repository local

Sumber : *https://idcloudhost.com/pengertian-dan-manfaat-git-bagi-developer/*

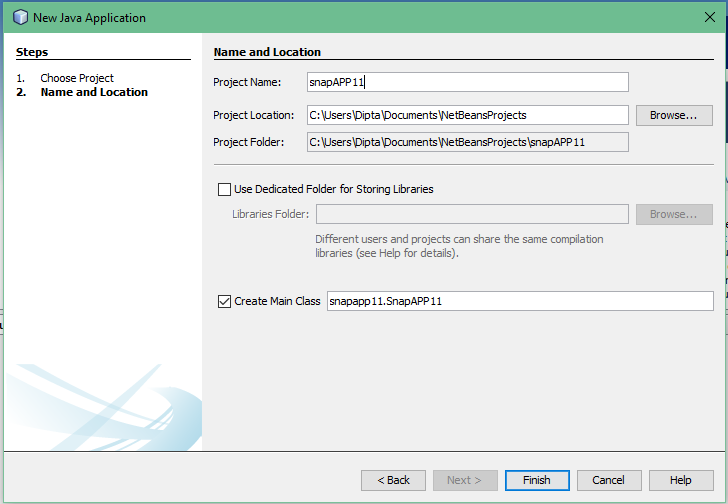
* 1. **Langkah Kerja**
     1. **NetBeans**

1. Buat file baru : File 🡪 New Project 🡪 Java 🡪 Java application



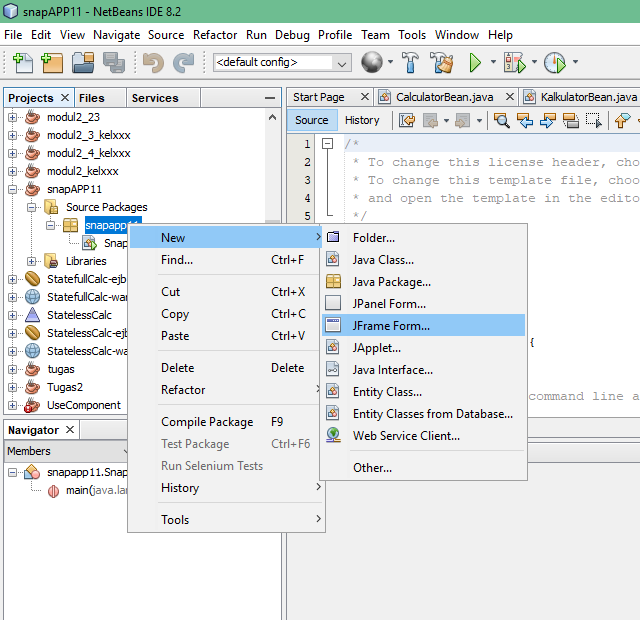
Gambar 2.6 Membuat project baru

1. Beri nama SnapApp11 (11 adalah kelompok )



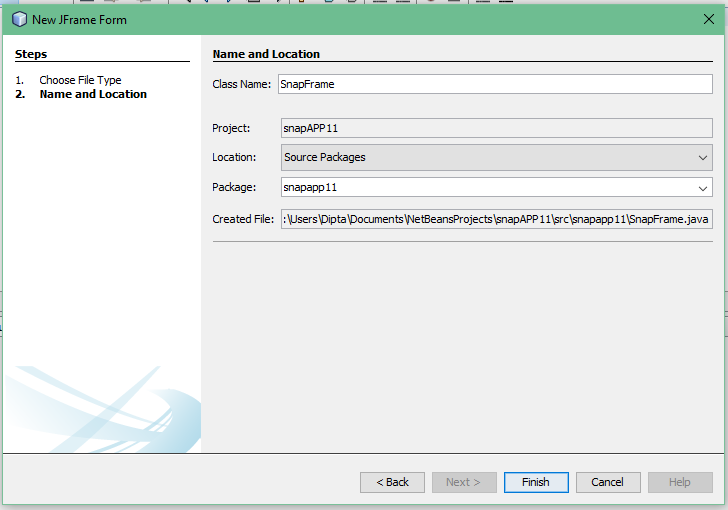
Gambar 2.7 Memberi nama project

1. Klik kanan pada package SnapApp 🡪 pilih New 🡪 pilih JFrame Form



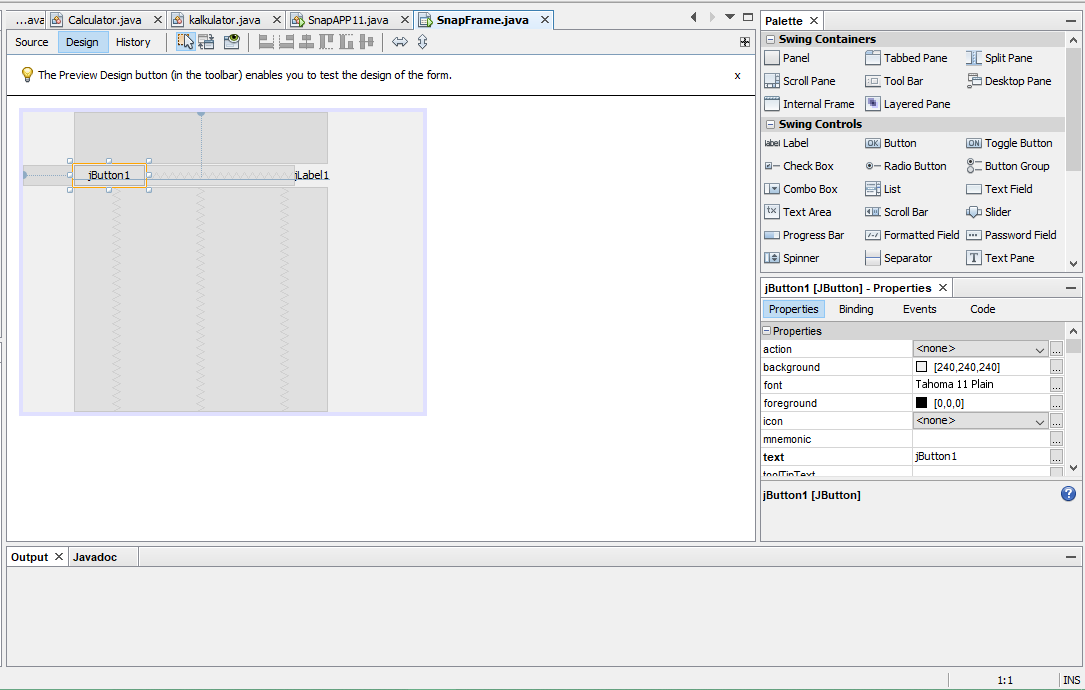
Gambar 2.8 Membuat JFrame Form

1. Beri nama SnapFrame



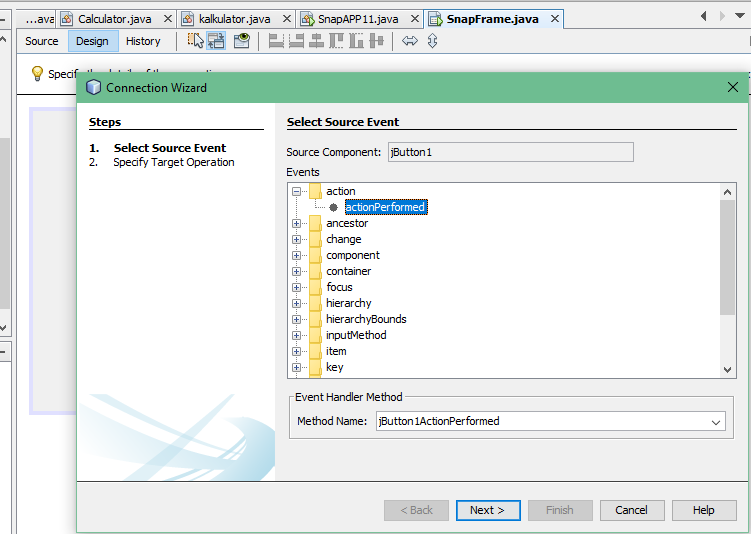
Gambar 2.9 Memberi nama JFrame Form

1. Buat tampilan seperti berikut ini



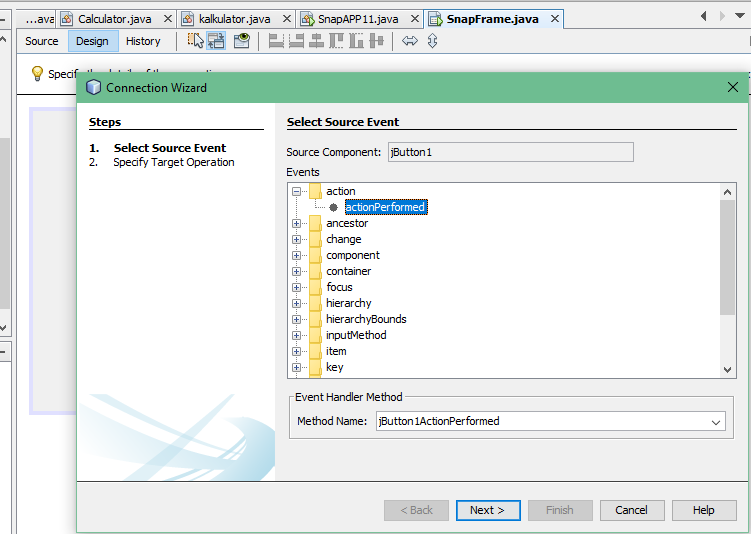
Gambar 2.10 Tampilan pada snapframe

1. Klik Connection Mode, kemudian klik pada tombol, kemudian klik pada label. Connection Mode digunakan untuk memberikan nilai pada label ketika button diberi action.



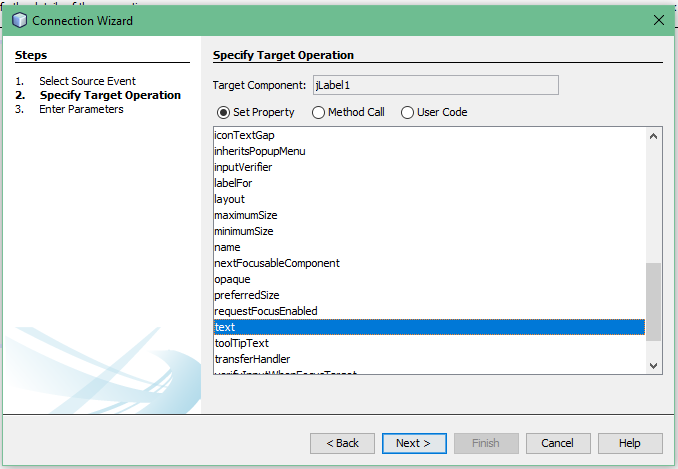
Gambar 2.11 Connection mode

1. Setelah itu akan muncul connection Wizard, pilih action 🡪 actionPerformed. Langkah ini akan membuat button ketika diberi action akan menjalankan events actionPerformed.



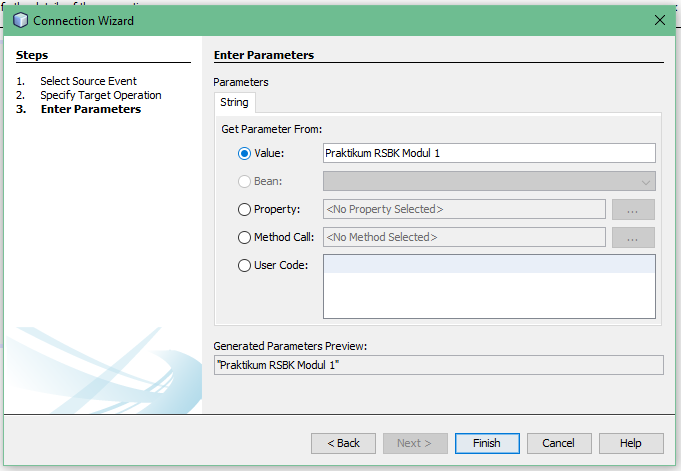
Gambar 2.12 Memilih action

1. Sekarang pengaturan pada labelnya, pilih set property 🡪 Text. Hal ini bertujuan untuk memilih action yang akan terjadi pada label.



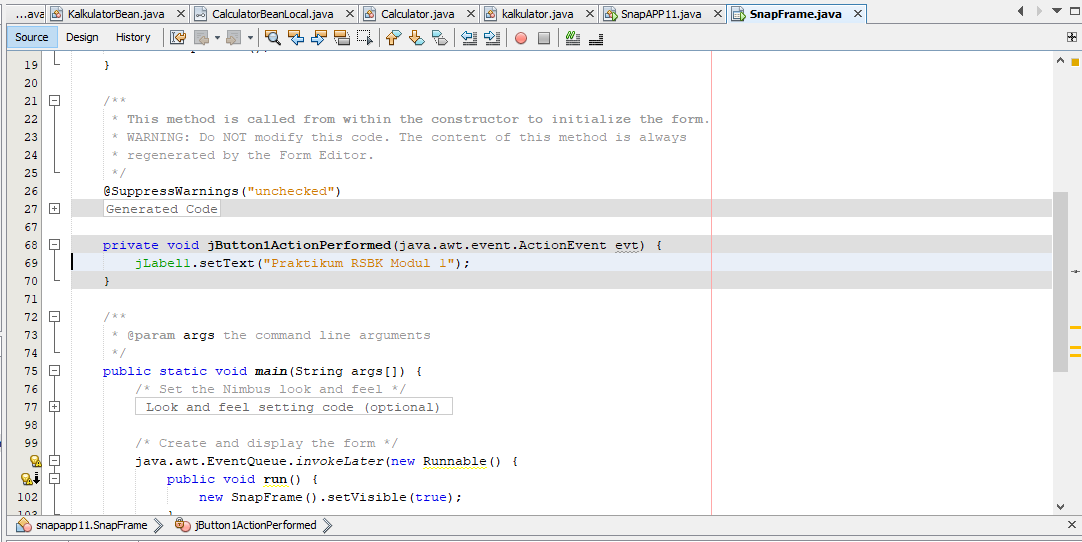
Gambar 2.13 Memilih ouput pada target component

1. Isi Value “praktikum rsbk modul 1”, kemudian finish. Ini akan menghasilkan output “praktikum rsbk modul 1” pada label ketika button diberi action.



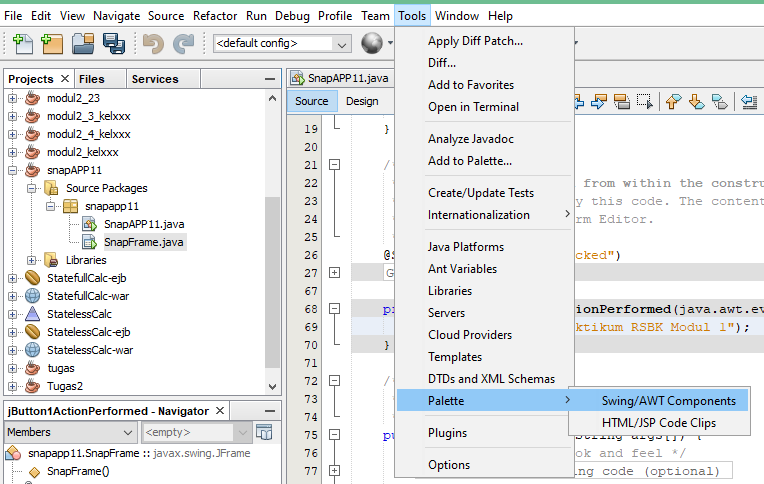
Gambar 2.14 Isi value label

1. Secara otomatis value akan muncul seperti source code berikut



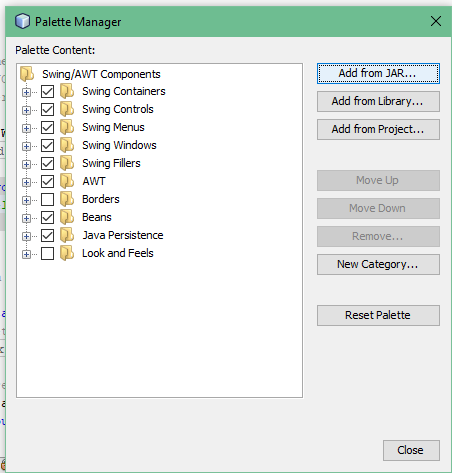
Gambar 2.15 Source code jButtonActionPerformed

1. Klik Tool 🡪 Pallete 🡪 Swing / AWT Components untuk mengimport component baru.



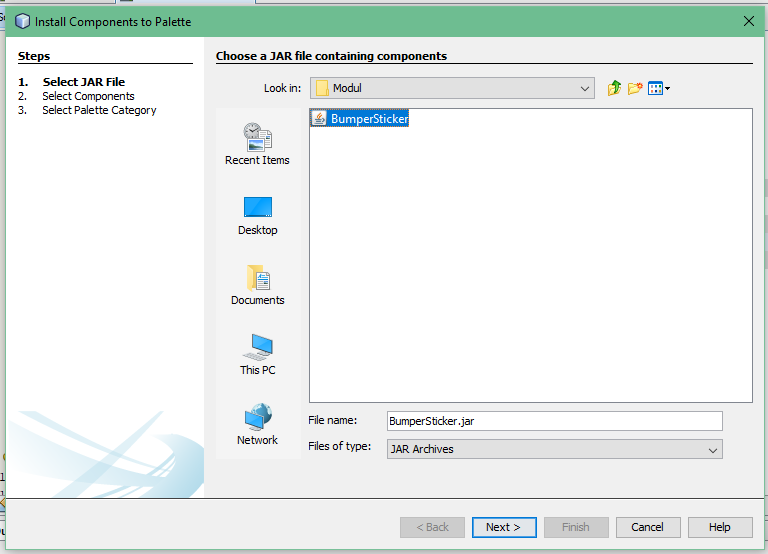
Gambar 2.16 Import beans

1. Klik Add From Jar untuk mulai memilih package component.



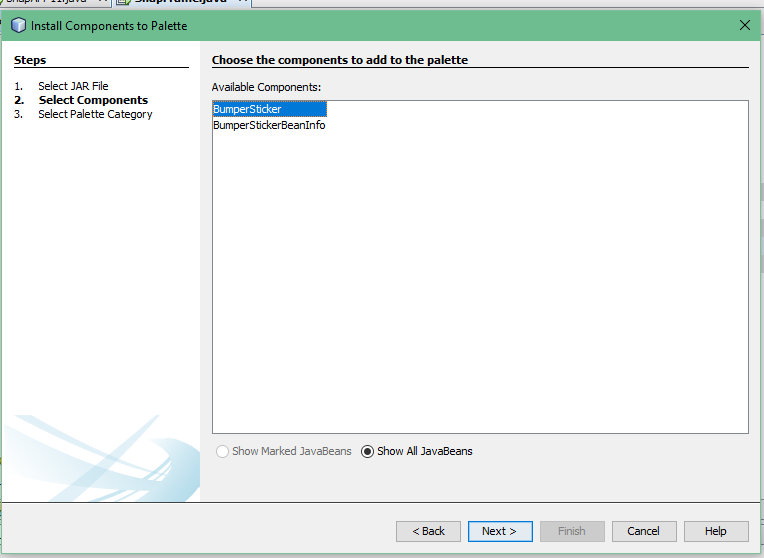
Gambar 2.17 Menambahkan component dari .jar yang sudah ada

1. Kemudian cari lokasi File BumperSticker, pilih filenya BumperSticker.jar



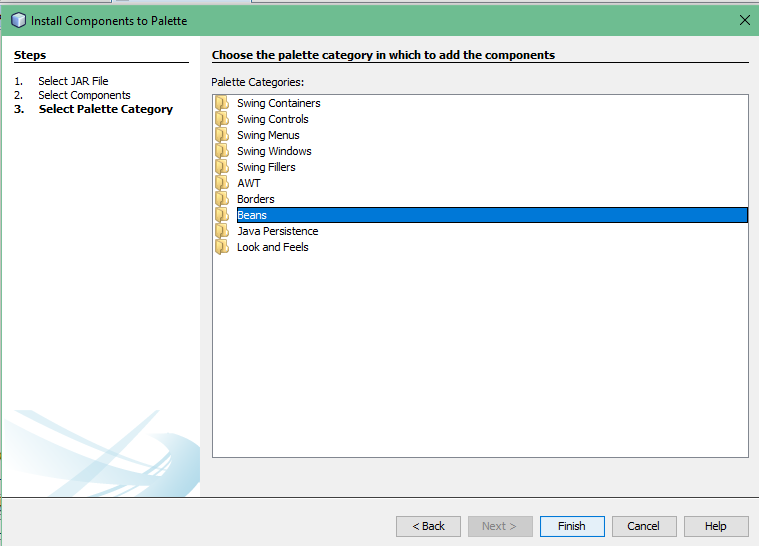
Gambar 2.18 Mencari lokasi bumperSticker.jar

1. Pilih bumperSticker



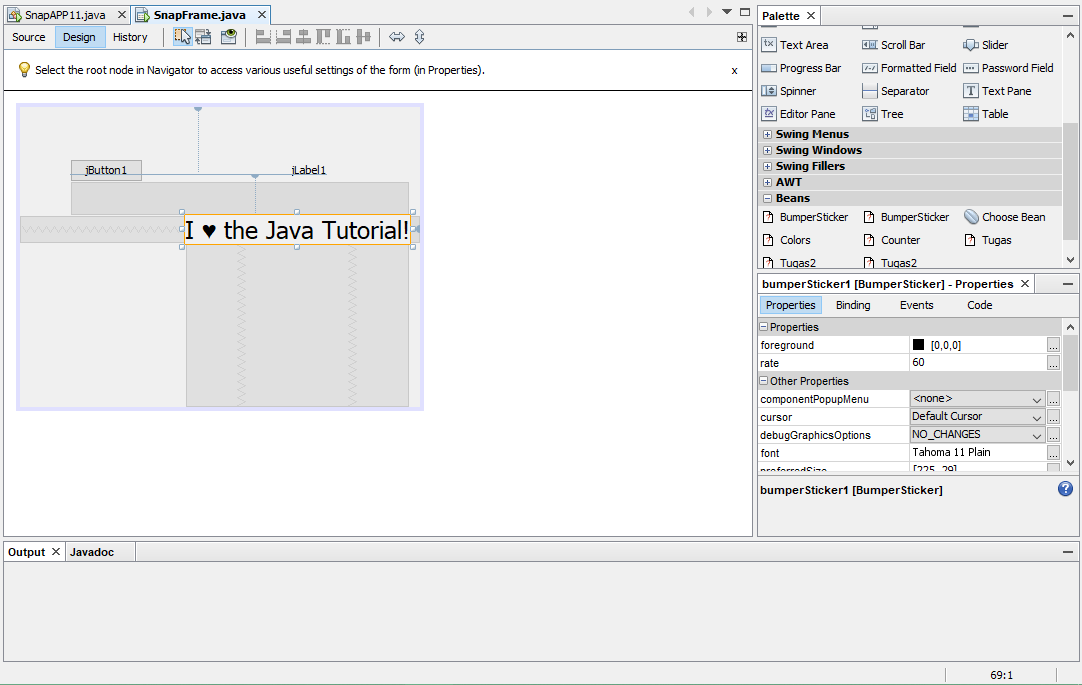
Gambar 2.19 Memilih bumperSticker

1. Pilih Beans yang berarti meletakkan component baru tersebut ke folder beans.



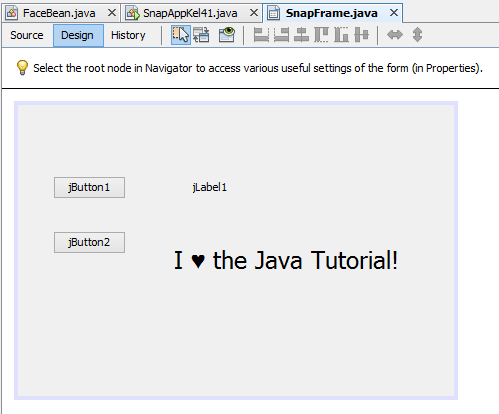
Gambar 2.20 Memilih folder tujuan

1. Drag Beans 🡪 BumperSticker ke Form



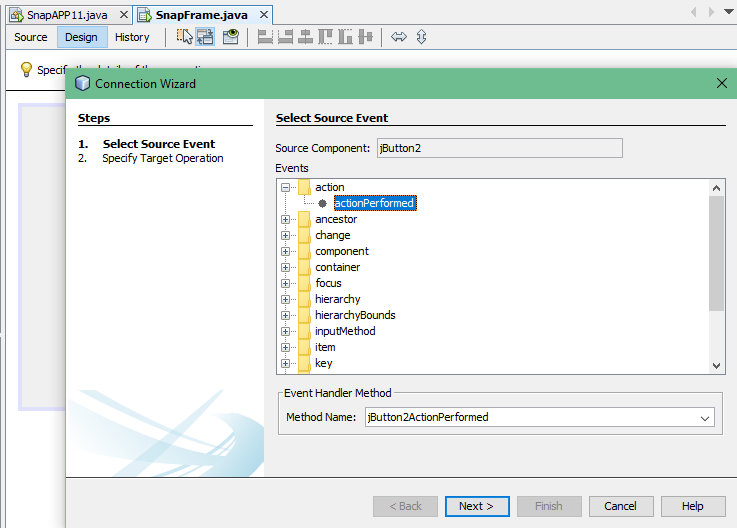
Gambar 2.21 Menambahkan bumpersticker ke form

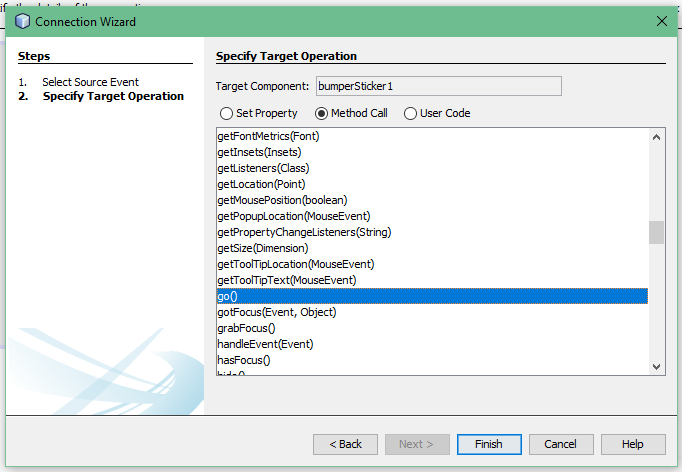
1. Tambahkan satu button lagi untuk menjalankan animasinya,



Gambar 2.22 Menambahkan button

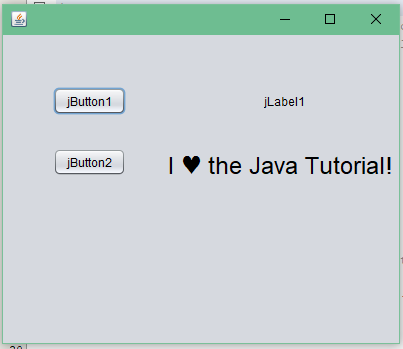
1. Koneksikan Button 2 dengan Bean BumperSticker, gunakan action performed, pilih MethodCall, pilih method go()



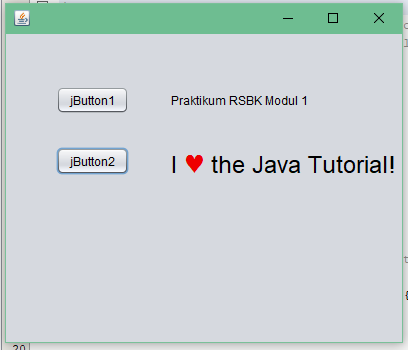


Gambar 2.23 Connection mode antara bumpersticker dan button

1. Setelah itu run projectnya. Jika jButton1 ditekan maka label akan menampilkan teks “praktikum rsbk modul 1” dan jika jButton2 ditekan maka gambar love akan berkedip merah hitam.



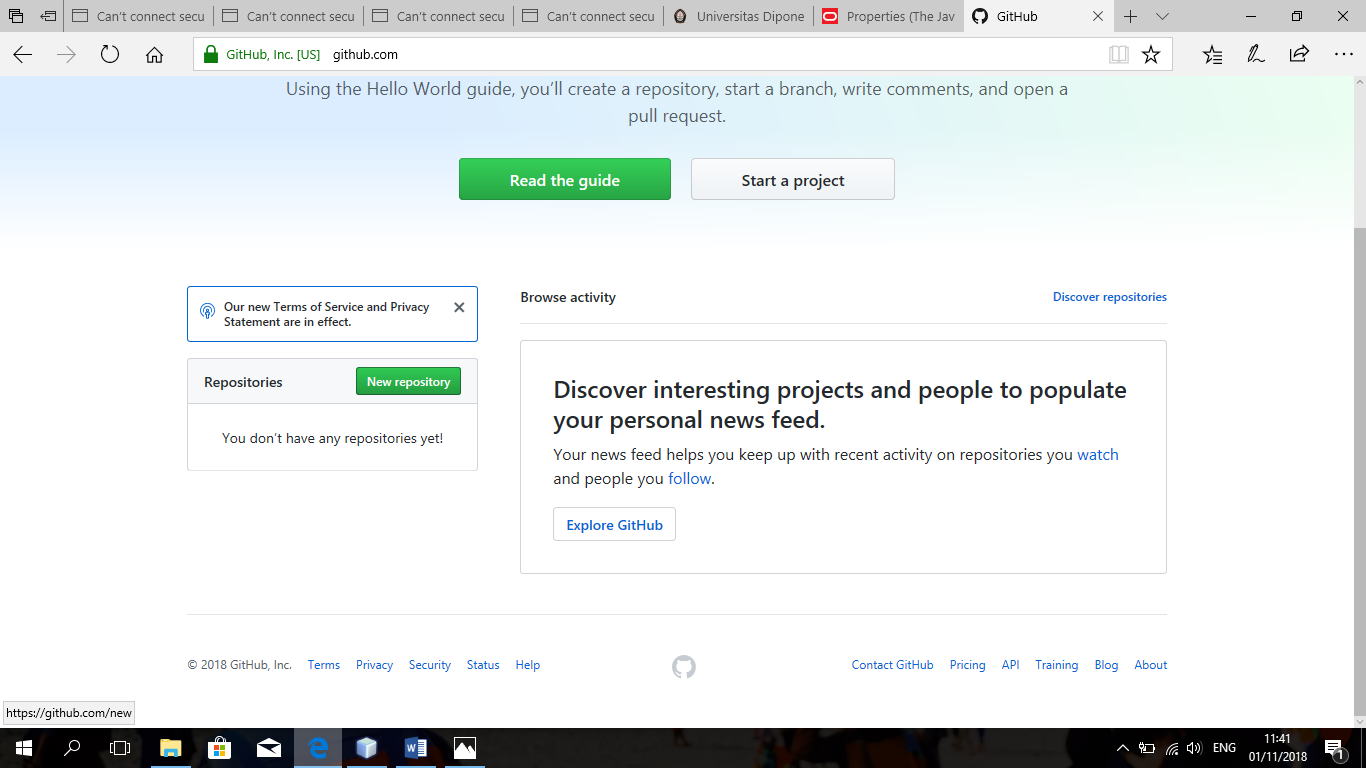
Gambar 2.24 sebelum di klik button



Gambar 2.25 setelah di klik button

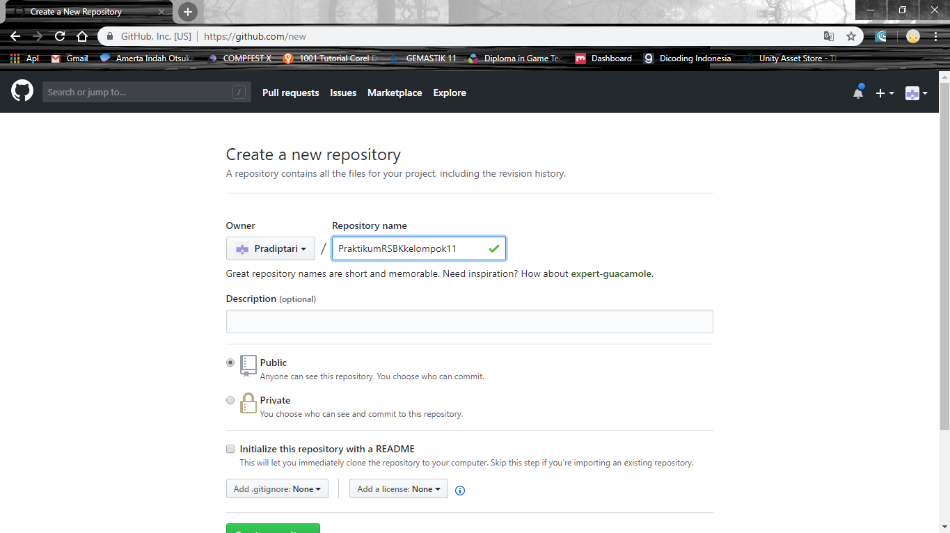
* + 1. **Git And GitHub**

1. Buka website github di www.github.com pada browser dan login sesuai dengan akun masing masing, kemudian Pilih tombol new yang terletak di sebelah kiri atas di dashboard masing masing akun untuk membuat repository pada akun github masing masing.



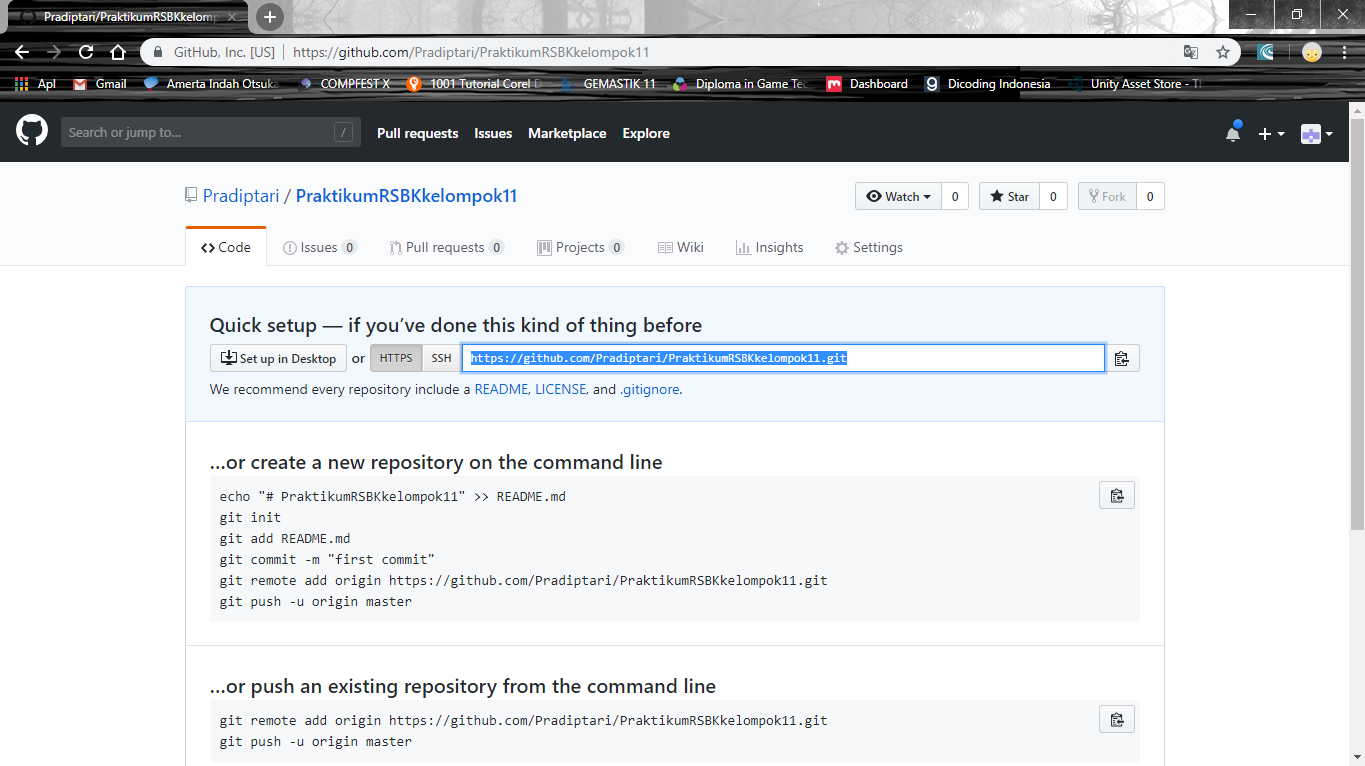
Gambar 2.26 new repository

1. Beri nama repository tersebut dengan nama PraktikumRSBKkel11 (11 = nomer kelompok), kemudia klik create repository



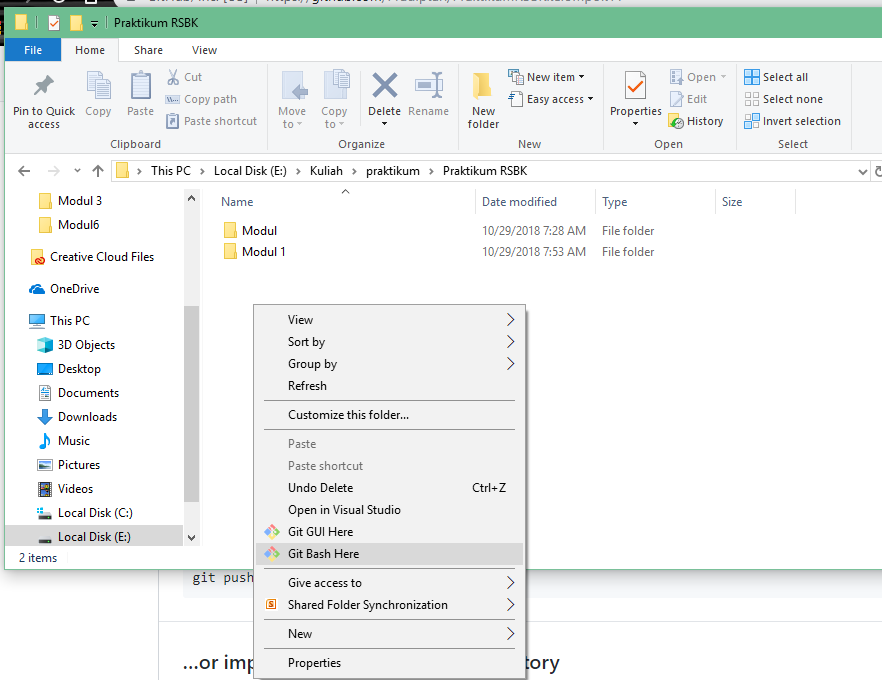
Gambar 2.27 Create repository

1. Setelah membuat repository maka repository tersebut siap diisi, copy link https yang ada di repository tersebut untuk dijadikan remote link yang nantinya dipakau untuk mengakses dari git.



Gambar 2.28 copy link yang ada di repository

1. Buka folder project java bean yang sudah dibuat tadi, kemudian klik kanan dan pilih opsi git bash untuk membuka console git.



Gambar 2.29 tampilan klik git bash

1. lalu ketikan beberapa perintah CLI, seperti berikut:

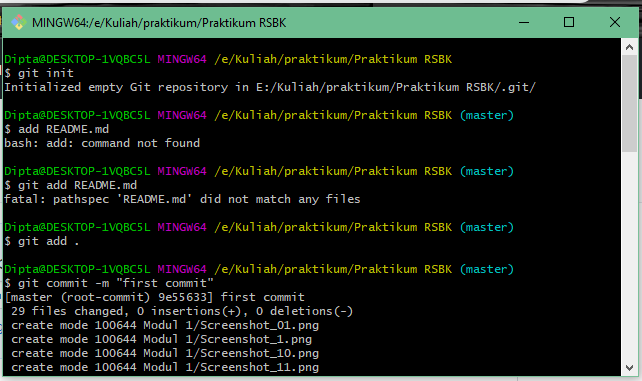
Git init

Git add .

Git commit –m “comment”

Git remote add origin link repository yang sudah copy

Git push origin master



Gambar 2.30 code untuk git init

Publish / init repository digunakan untuk mengunggah projek yang ada pada penyimpanan local atau computer kita ke web hosting github.com. Hal ini bertujuan untuk men-*track* perubahan atau untuk mem-*backup* file dan lain – lain. Untuk mengunggah repository ke web hosting github klik kanan pada folder project yang ingin diunggah.

Keterangan :

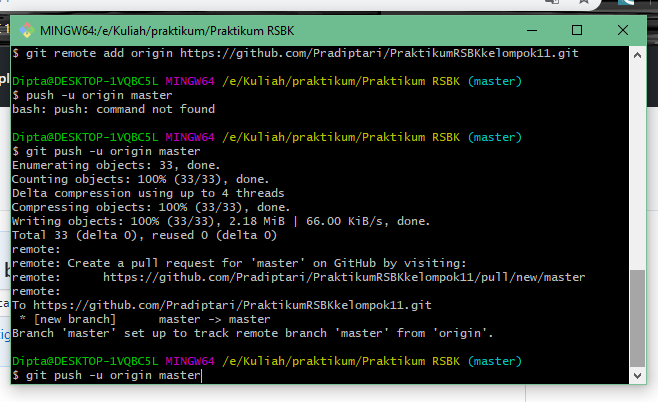
* + - * 1. “comment” dapat diganti dengan keterangan tentang apa yang diganti pada project
        2. Origin merupakan nama dari branch

1. Setelah repository berhasil di upload bisa di lihat di akun github masing masing bahwa repository di github sudah terisi dan terhubung dengan local repository yang sudah di buat tadi.
2. Kemudian untuk melakukan perubahan bisa dilakukan dengan cara update repository yang digunakan untuk mengubah code atau file pada repository kita yang ada di github kita sehingga sama dengan repository local, caranya hampir sama dengan saat kita menginisialisasi repository. Namu diawali dengan langkah add file. Langkahnya sebagai berikut :

Git add .

Git commit –m “comment”

Git push origin master



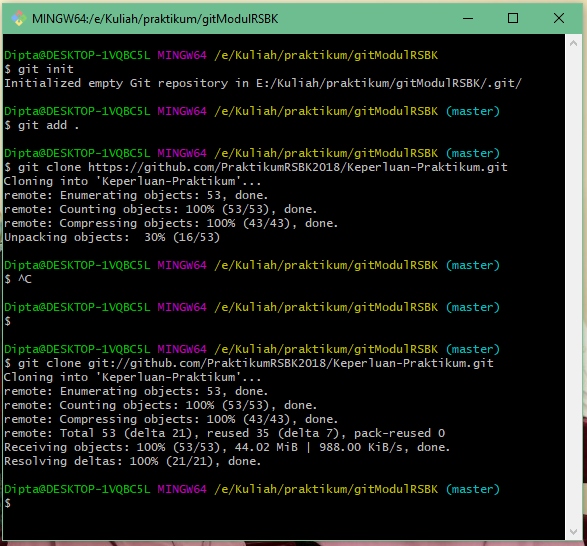
Gambar 2.31 code untuk git push

1. Perintah Pull Digunakan untuk mengupdate repository local sehingga sama dengan repository yang ada pada github. Caranya cukup mudah yaitu dengan cara :

Git pull

1. Terahir jika ingin mendownload / menggunakan repository orang lain yang ada di github cuku dengan copy link dari repository yang diinginkan bisa menggunakan Clone

Clone repository bertujuan untuk mengunduh sebuah projek yang ada pada github ke computer atau penyimpanan local, sehingga nantinya dapat diedit sesuai dengan keinginan sendiri. Untuk meng-*clone* repository git cukup ketikan perintah **git clone [url repository]**



Gambar 2.32 code untuk git clone

Directory dapat diubah tergantung dari dimana kalian mengklik kanan dan membuka bash, misal membuka gitbash di folder D:\contoh. Maka repository akan berada di D:\contoh\[nama repository]

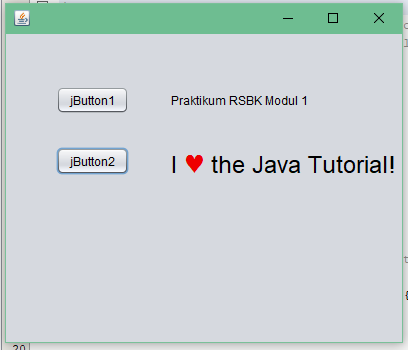
* + 1. **Analisis Percobaan**

Pada percobaan modul 1 ini . kita membuat SnapApp dengan menggunakan BumperStiker.jar pada netbeans. Komponen yang digunakan adalah 2 buah button dan 1 buah label. Button 1 akan dihubungkan dengan label 1 dan button2 akan dihubungkan dengan BumperStiker. Untuk menghubungkan Button1 dan label 1 klik pada Button (Jbutton1) dan label (jLabel1) di lembar kerja , kemudian akan muncul Connection Wizard. Kemudian pilih event yang kita inginkan. Untuk Button pilihlah action Performed. Klik next. Kemudian kita tentukan apa yang akan terjadi bila Button yang telah kita buat sebelumnya ditekan. Pilih text pada list, kemudian klik next. Isi value yaitu, Praktikum RSBK modul 1, kemudian klik finish. Untuk menghubungkan Button2 dan BumperSticker pilih connection mode ,kemudian klik pada jButton2 dan BumperSticker, kemudian akan muncul connection Wizard, klik tanda tambah disamping tulisan ‘action’ dan pilih actionperformed kemudian klik next pilih method call kemudian pilih go() dari list dan klik finish.

Setelah itu kita melakukan push file ke repository github , pertama kita buat repository baru pada akun github dan create repository baru untuk membuat folder pada github, kemudian akan mendapatkan link untuk di jadikan remote link pada saat mengakses git, setelah itu kita melakukan input code untuk menambahkan file pada repository github dengan perintah berikut Git remote add digunakan untuk menambahkan file yang akan di upload , dan origin adalah merupakan nama dari branch, dan link repository digunakan untuk remote link pada saat mengakses git, kemudian perintah untuk Git push yang berfungsi untuk memperbarui repository.

Dan untuk code Git pull Digunakan untuk mengupdate repository local sehingga sama dengan repository yang ada pada github, dan untuk Git clone digunakan untuk mendownload / menggunakan repository orang lain yang ada di github.

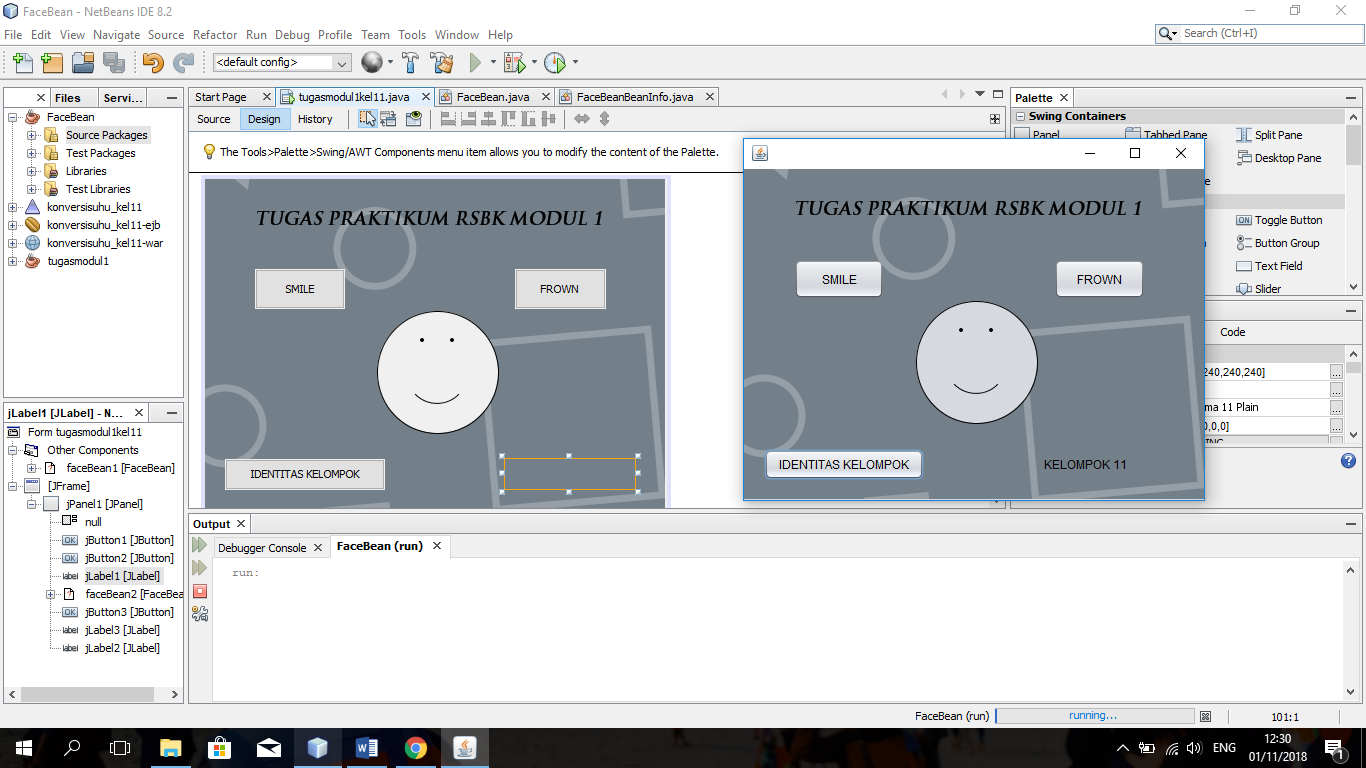
Hasil dari percobaan tersebut adalah jika Button1 dan Button2 ditekan akan muncul tampilan seperti gambar di bawah ini.



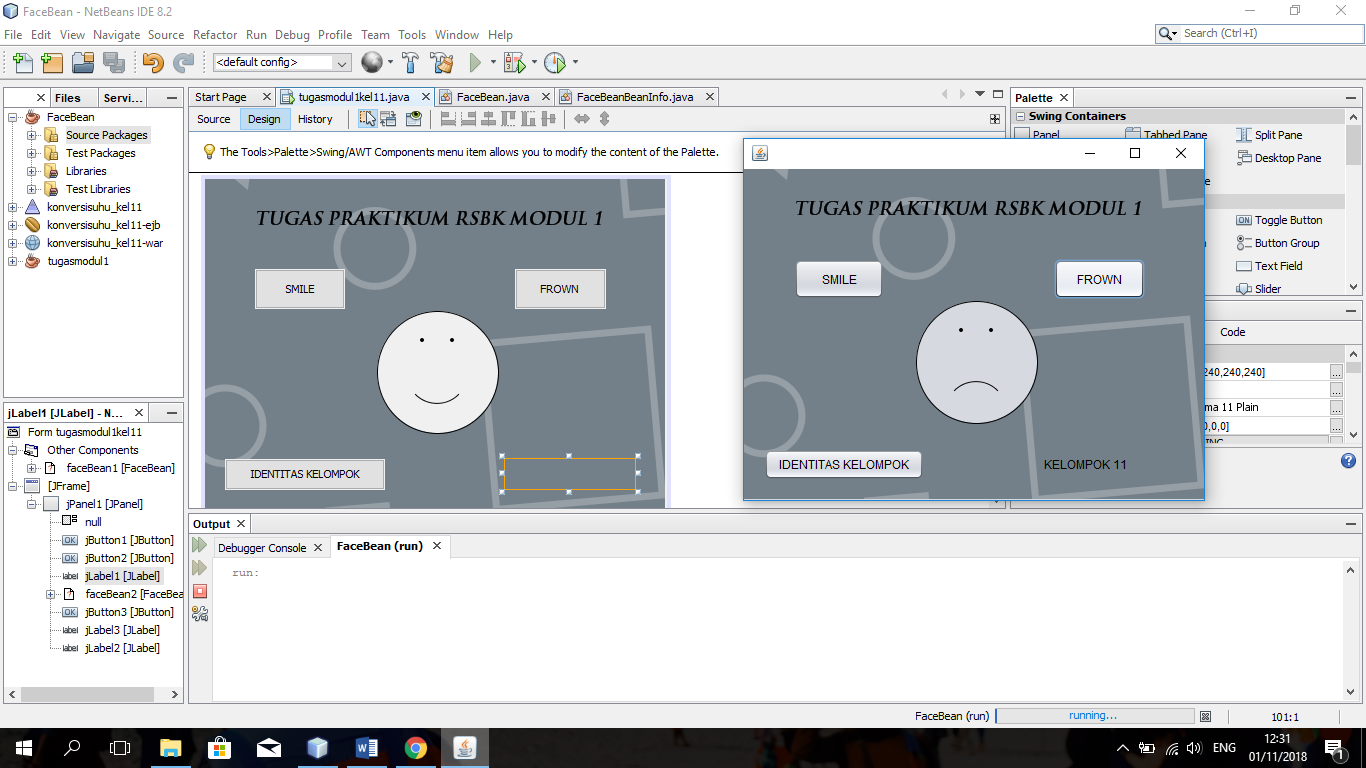
Gambar 2.33 hasil running dan setelah kedua button di klik

**2.5 Tugas dan Pembahasan**

Pada tugas modul 1 sama seperti langkah percobaan modul 1 ini . kita membuat facebean dengan menggunakan facebean.jar pada netbeans. Komponen yang digunakan adalah 3 buah button dan 1 buah label. Button 1 dan button2 akan dihubungkan dengan facebean.jar sedangkan Button 3 akan dihubungkan dengan label 1 Untuk menghubungkan Button1,Button 2 dan facebean.jar pilih connection mode ,kemudian klik pada jButton1 dan facebean kemudian akan muncul connection Wizard, klik tanda tambah disamping tulisan ‘action’ dan pilih actionperformed kemudian klik next pilih method call kemudian pilih smile() dari list dan klik finish. Pada Button 1 menambahkan fungsi method smile untuk memanggil perintah fungsi smile jika dia bernilai true, sedangkan Pada Button 2 menambahkan fungsi method frown untuk memanggil perintah fungsi frown jika dia bernilai true, kemudian Untuk menghubungkan Button3 dan label 1 klik pada Button (Jbutton3) dan label (jLabel1) di lembar kerja , kemudian akan muncul Connection Wizard. Kemudian pilih event yang kita inginkan. Untuk Button pilihlah action Performed. Klik next. Kemudian kita tentukan apa yang akan terjadi bila Button yang telah kita buat sebelumnya ditekan. Pilih text pada list, kemudian klik next. Isi value yaitu, kelompok 11, kemudian klik finish



Gambar 2.34 hasil running dan setelah klik button smile dan identitas kelompok



Gambar 2.35 hasil running dan setelah klik button frown dan identitas kelompok

Link tugas : (https://github.com/Pradiptari/PraktikumRSBKkelompok11.git)

**2.7 Kesimpulan**

1. Ada 3 mekanisme pada penambahan beans, yaitu menambahkan dari .jar, menambahkan dari library, dan menambahkan dari project.
2. Selain mengimport project beans yang sudah dibuild menjadi jar, dapat juga mengimport beans yang masih berupa project.
3. BumperSticker.jar merupakan third-party bean yang kita gunakan ini tidak disediakan secara default oleh NetBeans.
4. Connection mode berguna untuk mempermudah pengaturan event dengan diberi action akan menjalankan events action performed dibandingkan dengan menulis source code secara manual.
5. Semua komponen yang terdapat pada bagian pallete merupakan bean yang sudah tersedia secara default di NetBeans.
6. Untuk menambah pallete masuk ke windows tools-pallete.
7. Pada Git clone dapat digunakan untuk mendownload file secara keseluruhan juga dapat digunakan untuk melihat repository orang lain yang ada di github.
8. Github telah menjadi *social network* yang telah menyambungkan issue tracking dengan source control

**Ketentuan Asistensi:**

Subjek “Laporan\_RSBK\_BAB II\_KelompokXX”

Shift 1 : ACC ke Adi ([madinugroho@student.ce.undip.ac.id](mailto:madinugroho@student.ce.undip.ac.id)) Line : @adinugi / 081212583838

Shift 2 & 3 : ACC ke Bima ([bimafajarsetiawan@gmail.com](mailto:bimafajarsetiawan@gmail.com)) Line : @bima\_iero / 089610687965

Deadline ACC 6 November 2018