

# Buku Panduan Materi Workshop

---

Membuat Web di Java Seperti Swing Menggunakan

# vaadin }>

Iesa Maliek



Buku ini bebas untuk dicopy / didistribusikan secara gratis.  
Dilarang menggunakan sebagian / keseluruhan isi untuk tujuan komersial tanpa seijin penulis!  
Non Commercial use only. Copyleft 2010

## Sebuah Perkenalan

Halo, apa kabar semuanya? Selamat berjumpa dengan saya \*\*\*\*\* (nama disamarkan untuk menjaga privasi :-D). Bagaimana kabar kalian, apakah baik-baik saja? Hmm... kalau saya lihat dari ekspresi kalian, sepertinya sedang tidak dalam kondisi kejiwaan yang baik. Ada masalah apa kawan? Ayo katakan saja mungkin saya bisa membantu.

Ouw... Jadi masalah kalian itu ingin membangun aplikasi web tapi tidak mengerti apa itu HTML, JavaScript, JSP ataupun Servlet? Innalillahi wa innailaihi rooji'uun... penyakit kalian sungguh kronis kawan, sangat akut dan membahayakan. Penyakit no.21 paling berbahaya di Madiun (ups... bilang Medan kok Madiun – kebanyakan nonton Syahrini :-D), ini merupakan penyakit lama turunan mahasiswa IT Indonesia, yaitu “*nggak mau susah*” atau kalau kata bang Haji Rhoma istilah yang lebih ngetrendnya “*terleeena*” :-D (efek terlalu banyak curhat di facebook, DotA, warcraft, PB, dkk).

Setelah mengarungi benua Google, menyeberangi lautan Blog dan mengelilingi bukit Apache, saya menemukan sesuatu yang menarik yang akan saya bagi untuk kalian. Semoga saja setelah mempelajarinya kalian tidak lagi “*terleeena*”. Tenang kawan, yang satu ini noHTML, noJSP, dkk. Sangat cocok buat kalian yang “*nggak mau susah*”.

Kalaupun nantinya setelah mempelajari buku ini kalian masih juga belum bisa membuat web, masih juga nggak ngedong, paling tidak kalian sudah berusaha. Dan paling tidak

kalian akhirnya tahu dan seharusnya sadar diri bahwa ternyata kampus sempit Informatika bukan tempat yang tepat bagi kalian. Lekas kemasi barang dan pindah jurusan saja :-D (cepat-cepatlah bertobat anak muda, semoga lekas dikembalikan ke jalan yang lurus).

Demikianlah sebuah pengenalan yang menjadi awal perjumpaan kita, semoga yang akan saya sampaikan ini bermanfaat. Selamat mencoba dan **Happy Coding** ;-).

## -----== DAFTAR ISI ==-----

### (I) MEMPERKENALKAN VAADIN

Apa Itu Vaadin

Mengapa Vaadin

Kapan Menggunakan Vaadin

HelloVaadin

### (II) KOMPONEN USER INTERFACE

Label

Panel

Link

TextField

RichTextArea

Date dan Time

Button

OptionGroup

Select

Table

Tree

MenuBar

Form

### (III) LAYOUT VAADIN

VerticalLayout

HorizontalLayout

GridLayout

### (IV) DATABASE

Vaadin + MySQL

### (V) STUDI KASUS

Membuat JWebTranslator

## **(I) Memperkenalkan Vaadin**

### **Apa Itu Vaadin**

Pernahkah kalian belajar membuat web di Java? Ketika belajar kalian pasti bertemu dengan HTML/JSP, Servlet, CSS dan AJAX. Wow, sangat banyak sekali setidaknya ada 4 hal yang harus dikuasai. Tidak mudah memang, apalagi orang seperti saya yang dari awal belajar Java sudah terbiasa dengan Swing, pasti sangat sulit untuk belajar 4 hal itu.

*Tidak adakah yang lebih mudah dari ini? Mengingat saya lebih familiar dengan Swing.* Tenang jangan khawatir, ada framework Java yang sanggup membuat web semudah membuat Swing, namanya Vaadin. Tidak hanya itu saja Vaadin merupakan web framework yang digunakan untuk menghasilkan tampilan web yang lebih rich / lebih sejuk dipandang mata.

### **Mengapa Vaadin**

Vaadin memudahkan kita yang sudah sangat familiar dengan Swing untuk membuat aplikasi web. Kita tidak perlu lagi belajar sesuatu yang baru seperti HTML, CSS dan AJAX. Tentu saja hal ini sangat menghemat waktu belajar kita.

Sebenarnya ada teknologi lain selain Vaadin yang menyediakan kemudahan yang sama. Seperti : GWT, ZK, Wicket, dll. Lalu kenapa harus Vaadin yang dipilih? Sekarang coba kita bandingkan satu-persatu.

Kita mulai dari GWT. GWT menawarkan kemudahan dan sangat mirip dengan Vaadin, karena memang Vaadin dibuat dari GWT. Akan tetapi GWT membutuhkan compile terlebih dahulu menggunakan GWT compiler sebelum akhirnya dapat digunakan, sedangkan Vaadin tidak perlu. Inilah yang menyebabkan GWT menjadi kurang nyaman.

ZK sebetulnya sangat bagus dan mudah digunakan. Akan tetapi ZK memiliki license yang kurang bersahabat “**Free for Non-Commercial use**” (sebuah lisensi yang kurang nyaman bagi pencinta gratisan sejati :-D). Berbeda dengan Vaadin yang berlisensi **Apache 2.0** yang free untuk semua-muanya.

*Lalu bagaimana dengan Wicket, Wicket kan berlisensi Apache, selain itu juga sangat mudah dan mirip dengan Swing, lalu apa masalahnya? Wicket tetap membutuhkan HTML untuk templatingnya, itu yang jadi masalah :-D.*

## **Kapan Menggunakan Vaadin**

Vaadin sangat cocok digunakan untuk membuat aplikasi web yang berbau RIA (Rich Internet Application). Gampangnya mirip aplikasi desktop yang diweb-kan, tapi tidak dalam bentuk applet yang lambat dan menyusahkan itu.

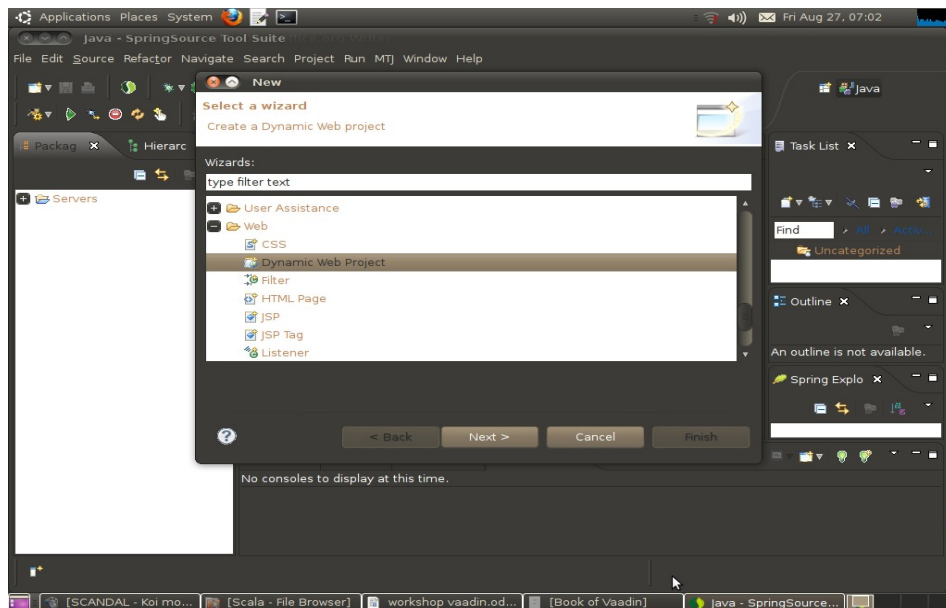
Meskipun banyak kemudahan, adakalanya Vaadin kurang cocok digunakan. Yaitu saat kita membuat web yang tidak membutuhkan banyak komponen seperti weblog atau website biasa yang tidak membutuhkan tampilan yang aneh-aneh. Untuk yang satu ini

HTML, dkk wajib digunakan, jadi jangan melupakan HTML :-D.

## HelloVaadin

Saatnya kita mulai mencoba Vaadin. Sebelumnya jangan lupa download dulu Vaadin nya di <http://www.vaadin.com> – downloadlah yang sesuai dengan OS yang kalian gunakan. Selain itu siapkan juga Apache Tomcat untuk menjalankan web yang akan kita buat. Yang tidak kalah penting siapkan Java IDE untuk mempermudah pekerjaan kita, bisa pakai Netbeans atau Eclipse untuk JavaEE. Selanjtnya akan saya demokan satu-persatu.

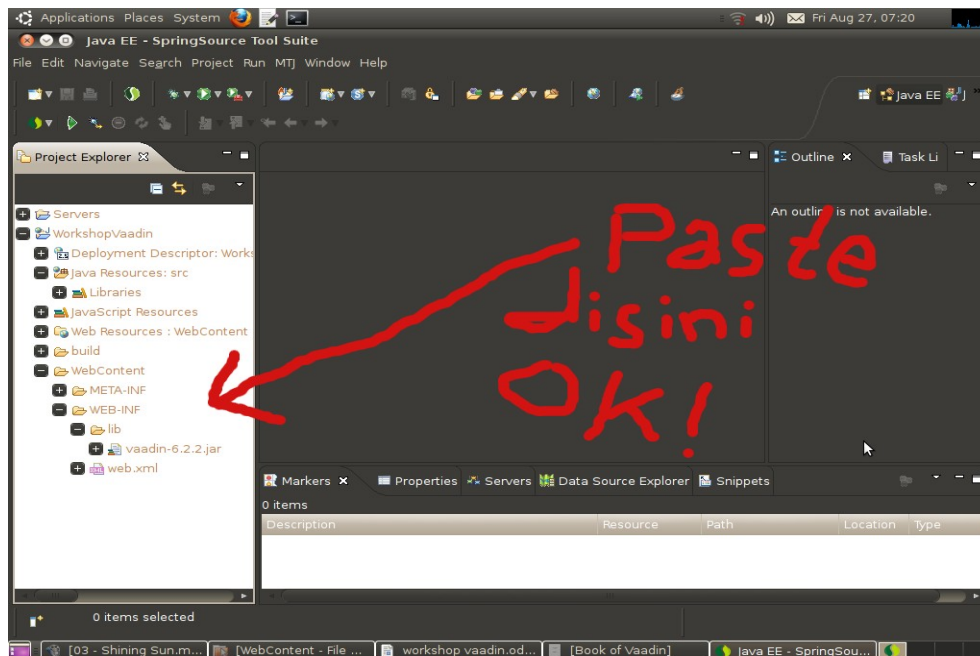
Untuk yang pakai Eclipse, seperti biasa kita buat project dahulu sebelum memulai dan pilih “dynamic web project”.



Beri nama project nya dengan nama “WorkshopVaadin”, trus klik “Next” sampai “Finish”. Project telah siap kita gunakan. Setelah itu copy library Vaadin ke dalam project – setelah didownload ekstrak Vaadin dan buka foldernya cari file vaadin-6.x.x.jar file inilah yang akan kita pakai. Copy dan paste ke dalam folder “lib” yang ada didalam



project “WorkshopVaadin/WebContent/WEB-INF/lib”.



Selanjutnya kita buat Java class. Klik kanan di folder “src” dan pilih “New → Class”.

Beri nama class nya “HelloVaadin” dan beri nama package nya dengan

“workshop.vaadin”. Kita sudah dibuatkan class baru oleh Eclipse dan sekarang edit class tersebut sehingga menjadi seperti dibawah ini :

```
package workshop.vaadin;

import com.vaadin.Application;
import com.vaadin.ui.Label;
import com.vaadin.ui.Window;

public class HelloVaadin extends Application{
    @Override
    public void init() {
        Window main = new Window("Workshop Vaadin");
        setMainWindow(main);
        Label label = new Label("Hello Vaadin!");
        main.addComponent(label);
    }
}
```

Buka file web.xml yang ada didalam project dan edit menjadi seperti berikut :

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
```

```

http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
id="WebApp_ID" version="2.5">

<display-name>WorkshopVaadin</display-name>

<context-param>
  <description>Vaadin production mode</description>
  <param-name>productionMode</param-name>
  <param-value>false</param-value>
</context-param>

<servlet>
  <servlet-name>Workshop Vaadin</servlet-name>
  <servlet-class>
    com.vaadin.terminal.gwt.server.ApplicationServlet
  </servlet-class>

  <init-param>
    <description>Vaadin application class to start</description>
    <param-name>application</param-name>
    <param-value>
      workshop.vaadin.HelloVaadin
    </param-value>
  </init-param>
</servlet>

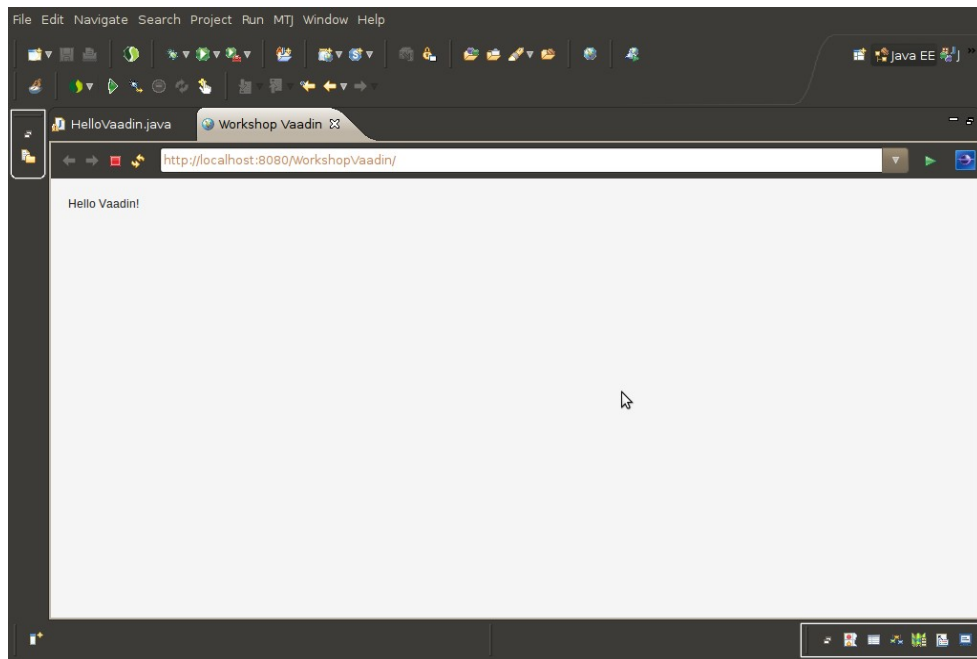
<servlet-mapping>
  <servlet-name>Workshop Vaadin</servlet-name>
  <url-pattern>/*</url-pattern>
</servlet-mapping>

</web-app>

```

Tunggu jangan buru-buru menjalankan, kita setting dulu AppServer yang akan kita gunakan untuk menjalankan. Klik menu “File → New → Other” pilih yang bagian “Server” kemudian “Next”. Pilih “Apaxe Tomcat” (pilihlah yang sesuai dengan versi Tomcat kalian) dan “Next”. Klik “Browse” dan cari tempat instalasi Tomcat kalian, setelah itu klik “Next” sampai “Finish”.

Untuk menjalankan klik kanan pada project “WorkshopVaadin → Run As → Run On Server” pilih Tomcat nya dan klik “Next” sampai “Finish”. Tunggulah beberapa saat dan hasilnya akan keluar. Sederhana memang hanya menampilkan tulisan “**Hello Vaadin**”, lumayan untuk mengawali eksplorasi kita.



Coba kita lihat kembali kode programnya bagaimana bisa menampilkan output yang seperti itu. Kalian yang terbiasa dengan Swing pasti sudah mengerti apa maksudnya.

```
Window main = new Window("Workshop Vaadin");  
setMainWindow(main);  
Label label = new Label("Hello Vaadin!");  
main.addComponent(label);
```

Coba kita bandingkan dengan Swing.

```
Jframe frame = new Jframe("Workshop Vaadin");  
Jlabel label = new Jlabel("Hello Vaadin!");  
frame.add(label);  
add(frame);
```

*Ya, memang persis sekali secara konsep koding, tapi apa itu maksud dari file web.xml?*

File web.xml itu berisi konfigurasi, ya memang ribet tapi tidak usah terlalu pusing cukup barisan kode ini saja yang perlu diperhatikan :

```
<param-value>  
    workshop.vaadin.HelloVaadin  
</param-value>
```

Itu adalah code wajib yang menunjukkan dimana letak class yang menjadi main aplikasi. Jadi kalau main aplikasinya diganti ke class lain, maka konfigurasi XML diatas juga harus diganti ke class main aplikasi yang baru. Bagaimana mudah bukan?

OK, karena sudah pada bisa kita akhiri saja pembahasan Vaadin sampai disini :-D

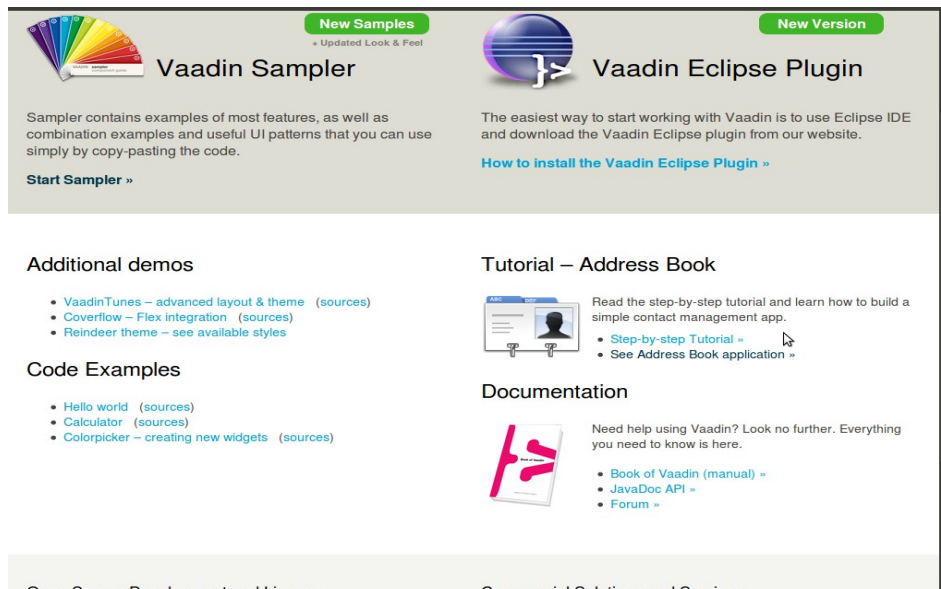
(penonton kecewa siap-siap mau lempar monitor dan CPU). Hehehe... tenang tidak perlu panik begitu, just kidding just for laugh.

Oh iya, ada yang kelupaan untuk Netbeans nya belum. OK, kita cobain untuk yang menggunakan Netbeans. Untuk Netbeans ini lebih mudah daripada Eclipse. Pertama-tama seperti biasa kita buat dulu project nya (Pilih yang Web Application dan ubah server nya menggunakan Tomcat). Setelah itu kita pasang library Vaadin ke project. Klik kanan pada project pilih “Properties → Libraries → Add JAR/Folder”. Cari library vaadin-6.x.x.jar kemudian klik “OK”. Setelah itu Buat Class baru dan isikan kodenya sama seperti kode diatas. Ubah juga web.xml nya sama persis seperti diatas tadi. Untuk menjalankannya sangat mudah tinggal klik kanan pada project dan pilih “Run”. Jreng..jreng.. hasilnya pun sama seperti yang dibuat dengan Eclipse.

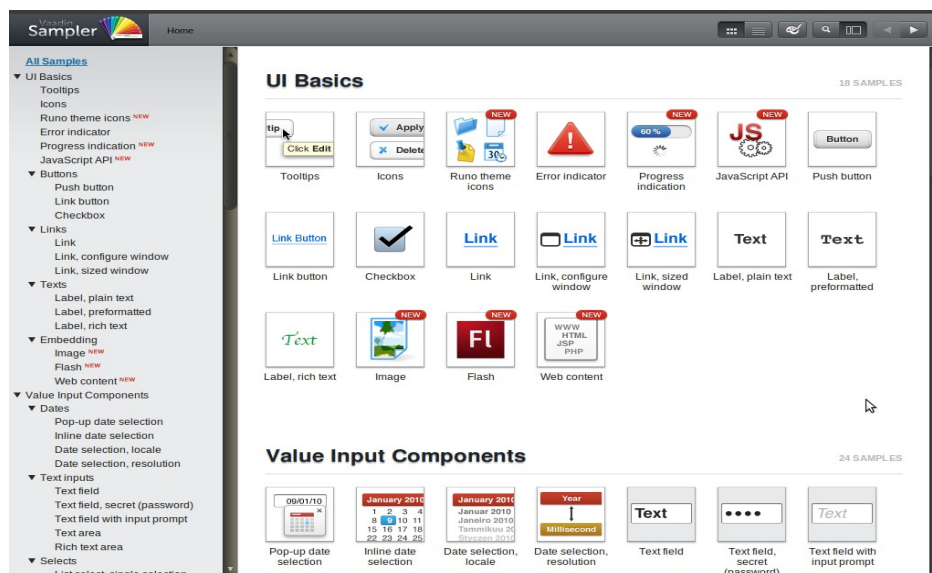
Didalam file distribusi Vaadin ada banyak dan lengkap contoh-contoh cara menggunakan Vaadin. Untuk melihatnya, buka **command prompt** pindahkan direktori ke tempat instalasi Vaadin, kemudian ketikkan perintah **sh start.sh** (linux) **start.bat** (windows) :

```
kakashi@kakashi:~$ cd Installed/vaadin-linux-6.2.2/
kakashi@kakashi:~/Installed/vaadin-linux-6.2.2$ sh start.sh
-----
Starting Vaadin in Demo Server.
Running in http://localhost:8888
-----
2010-09-04 07:38:19.047::INFO: Logging to STDERR via
org.mortbay.log.StdErrLog
2010-09-04 07:38:19.115::INFO: jetty-6.1.7
2010-09-04 07:38:20.310::INFO: NO JSP Support for , did not find
org.apache.jasper.servlet.JspServlet
2010-09-04 07:38:20.582::INFO: Started SelectChannelConnector@0.0.0.0:8888
```

Perintah diatas akan menjalankan Appserver Jetty dan secara otomatis akan menjalankan web browser terlihat seperti dibawah ini :



Untuk melihat contoh-contoh Vaadin klik pada gambar yang bertuliskan **Vaadin Sampler**. Inilah yang membuat saya senang banget sama Vaadin, selain dokumentasi yang baik dan dilengkapi dengan pdf, sample-sampelnya pun sangat lengkap dan disertai dengan source code. Nah, kalau suatu saat menemui kesulitan, bacalah dokumentasinya atau lihat sample source code vaadin untuk membantu kalian.



## **(II) Komponen User Interface**

Vaadin merupakan framework yang menyediakan komponen didalamnya (framework berbasis komponen). Komponen yang disediakan Vaadin sangat mirip dengan Swing, ini yang akan membuat kita mudah untuk mempelajarinya. Hmm... kalian masih belum percaya? OK, kita lihat satu-persatu.

### **Label**

Label merupakan komponen yang digunakan untuk menampilkan text yang tidak dapat diedit. Dibawah ini adalah contoh dari label.

```
Label label = new Label("Ini adalah label");
```

Label ini juga dapat kita modifikasi supaya dapat menampilkan text HTML. Berikut ini contohnya.

```
Label label = new Label("<h1>Ini adalah label</h1>", Label.CONTENT_XHTML);
```

Bagaimana canggih bukan.

### **Panel**

Panel merupakan sebuah wadah sederhana yang dapat menampung komponen lain didalamnya. Biasanya panel digunakan untuk halaman yang membutuhkan tampilan yang rumit dan membutuhkan banyak komponen.

Berikut adalah contoh panel yang menampung label didalamnya.

```
Panel panel = new Panel("Panel");
Label label = new Label("Ini adalah label");
panel.addComponent(label);
```

## Link

Link merupakan komponen yang memungkinkan kita untuk memanggil halaman lain baik itu halaman web yang berasal dari luar ataupun halaman web yang berasal dari aplikasi itu sendiri.

```
final Window mywindow = new Window("Window kedua");
mywindow.setName("windowdua");
mywindow.addComponent(new Label("Label dari window kedua"));
addWindow(mywindow);
Link link = new Link("Klik untuk membuka window yang kedua",
    new ExternalResource(mywindow.getURL()));
link.setTargetName("kedua");
link.setTargetHeight(300);
link.setTargetWidth(300);
link.setTargetBorder(Link.TARGET_BORDER_DEFAULT);
main.addComponent(link);
main.addComponent(
    new Label("Window kedua <a href='" + mywindow.getURL()
        + "' target='second'>klik untuk membuka</a>",
        Label.CONTENT_XHTML));
main.addComponent(
    new Label("Window kedua dapat diakses melalui URL: "
        + mywindow.getURL()));
```

## TextField

TextField merupakan komponen untuk editing yang paling umum dan dapat menyesuaikan diri dari berbagai macam kebutuhan. TextField dapat mendukung untuk editing single-line maupun multi-line (seperti JTextArea pada Swing) dan juga dapat digunakan sebagai input password (seperti JpasswordField pada Swing).

```
TextField single = new TextField("Single Line");
single.setColumns(40);
```

```
main.addComponent(single);

TextField multi = new TextField("Multi Line");
multi.setRows(7);
multi.setColumns(40);
main.addComponent(multi);

TextField password = new TextField("Password");
password.setSecret(true);
main.addComponent(password);
```

## RichTextArea

RichTextArea merupakan komponen yang memungkinkan kita untuk melakukan pengeditan text dengan format HTML. Dengan kata lain RichTextArea merupakan TextField yang memiliki kemampuan atau level yang lebih tinggi.

```
final RichTextArea rtarea = new RichTextArea();
rtarea.setCaption("Rich Text Area");
rtarea.setValue("<h1>Hello</h1>\n" +
                "<p>Selamat datang di workshop vaadin.</p>");
main.addComponent(rtarea);
```

## DateField

DateField merupakan komponen untuk meng-input dan menampilkan tanggal maupun waktu.

```
DateField date = new DateField("kalender DateField");
date.setValue(new java.util.Date());
main.addComponent(date);
```

## Button

Button merupakan komponen yang biasanya digunakan untuk menandai diawalinya suatu proses, misalnya login baru akan diproses setelah user menekan tombol OK.

```
Button button = new Button("Button");
button.addListener(new ClickListener() {
```



```

@Override
public void buttonClick(ClickEvent event) {
    getMainWindow().showNotification("hello","hello");
}
});

```

## OptionGroup

OptionGroup mirip dengan dapat memilih pilihan, namun hanya satu saja.

```

OptionGroup og = new OptionGroup("True False");
og.addItem("Islam");
og.addItem("Kristen");
og.addItem("Katolik");
og.addItem("Hindu");
og.addItem("Budha");
og.addItem("Konghuchu");

```

## Select

Select menyediakan pilihan alternatif untuk memilih satu atau lebih dari sebuah list.

```

Select select = new Select("Select something here");
main.addComponent(select);
final String[] planets = new String[] {
    "Mercury", "Venus", "Earth", "Mars",
    "Jupiter", "Saturn", "Uranus", "Neptune"};
for (int i=0; i<planets.length; i++)
    select.addItem(planets[i]);

```

## Table

Digunakan untuk menampilkan data yang terorganisasi ke dalam baris dan kolom.

```

Table table = new Table("This is my Table");
table.addContainerProperty("First Name", String.class, null);
table.addContainerProperty("Last Name", String.class, null);
table.addContainerProperty("Year", Integer.class, null);

table.addItem(new Object[] {
    "Nicolaus","Copernicus",new Integer(1473)}, new Integer(1));
table.addItem(new Object[] {
    "Tycho", "Brahe", new Integer(1546)}, new Integer(2));
table.addItem(new Object[] {
    "Giordano","Bruno", new Integer(1548)}, new Integer(3));
table.addItem(new Object[] {
    "Galileo", "Galilei", new Integer(1564)}, new Integer(4));

```

```

table.addItem(new Object[] {
    "Johannes", "Kepler",    new Integer(1571)}, new Integer(5));
table.addItem(new Object[] {
    "Isaac",    "Newton",    new Integer(1643)}, new Integer(6));
table.setPageLength(3);
table.setSelectable(true);

```

## Tree

Tree merupakan komponen yang digunakan untuk merepresentasikan data secara hirarki.

```

GridLayout gl = new GridLayout(2,1);
final Label label = new Label();
Panel p1 = new Panel("Tree");
p1.setHeight("200px");
Panel p2 = new Panel("Label");
p2.setHeight("200px");
p2.setWidth("100px");
final Object[][] planets = new Object[][]{
    new Object[]{"Mercury"},
    new Object[]{"Venus"},
    new Object[]{"Earth", "The Moon"},
    new Object[]{"Mars", "Phobos", "Deimos"},
    new Object[]{"Jupiter", "Io", "Europa", "Ganymedes",
        "Callisto"},
    new Object[]{"Saturn", "Titan", "Tethys", "Dione",
        "Rhea", "Iapetus"},
    new Object[]{"Uranus", "Miranda", "Ariel", "Umbriel",
        "Titania", "Oberon"},
    new Object[]{"Neptune", "Triton", "Proteus", "Nereid",
        "Larissa"}
};
Tree tree = new Tree();
for (int i=0; i<planets.length; i++) {
    String planet = (String) (planets[i][0]);
    tree.addItem(planet);
    if (planets[i].length == 1) {
        tree.setChildrenAllowed(planet, false);
    } else {
        for (int j=1; j<planets[i].length; j++) {
            String moon = (String) planets[i][j];
            tree.addItem(moon);
            tree.setParent(moon, planet);
            tree.setChildrenAllowed(moon, false);
        }
    }
}
tree.addListener(new ItemClickListener() {
    @Override
    public void itemClick(ItemClickEvent event) {
        label.setValue(event.getItemId()+" Clicked");
    }
});
p1.addComponent(tree);

```

```
p2.addComponent(label);
gl.addComponent(p1);
gl.addComponent(p2);
main.addComponent(gl);
```

## MenuBar

Seperti pada aplikasi berbasis desktop MenuBar digunakan untuk membuat drop-down menu secara horisontal.

```
final MenuBar menubar = new MenuBar();
main.addComponent(menubar);

final Label selection = new Label("-");
main.addComponent(selection);

MenuBar.Command mycommand = new MenuBar.Command() {
    public void menuSelected(MenuItem selectedItem) {
        selection.setValue("Ordered a " +selectedItem.getText()+" from menu.");
    }
};

MenuBar.MenuItem beverages =
    menubar.addItem("Beverages", null, null);
MenuBar.MenuItem hot_beverages =
    beverages.addItem("Hot", null, null);
hot_beverages.addItem("Tea", null, mycommand);
hot_beverages.addItem("Coffee", null, mycommand);
MenuBar.MenuItem cold_beverages =
    beverages.addItem("Cold", null, null);
cold_beverages.addItem("Milk", null, mycommand);

MenuBar.MenuItem snacks =
    menubar.addItem("Snacks", null, null);
snacks.addItem("Weisswurst", null, mycommand);
snacks.addItem("Salami", null, mycommand);

MenuBar.MenuItem services =
    menubar.addItem("Services", null, null);
services.addItem("Car Service", null, mycommand);
```

## Form

Yang paling menarik dari vaadin adalah Form. Form merupakan komponen yang didalamnya sudah terdapat layout yang ditujukan untuk pemakaian tertentu, misalnya untuk halaman login.

```
Form form = new Form();
form.setCaption("Form");
form.setDescription("Ini merupakan deskripsi dari form.");
```

```
form.getLayout().addComponent(new TextField("Field 1"));
form.addField("another", new TextField("Field 2"));
form.setFooter(new VerticalLayout());
HorizontalLayout okbar = new HorizontalLayout();
okbar.setHeight("25px");
form.getFooter().addComponent(okbar);
Button okbutton = new Button("OK", form, "commit");
okbutton.addListener(new ClickListener() {
    @Override
    public void buttonClick(ClickEvent event) {
        getMainWindow().showNotification("hello","hello");
    }
});
okbar.addComponent(okbutton);
okbar.setComponentAlignment(okbutton, Alignment.TOP_RIGHT);
okbar.addComponent(new Button("Reset", form, "discard"));
okbar.addComponent(new Button("Cancel"));
main.addComponent(form);
```

## (III) Layout Vaadin

### VerticalLayout

VerticalLayout merupakan sebuah layout yang akan mengatur letak komponen secara vertikal atau dari atas kebawah.

```
VerticalLayout vertikal = new VerticalLayout();
Button button1 = new Button("Button 1");
Button button2 = new Button("Button 2");
Button button3 = new Button("Button 3");
Button button4 = new Button("Button 4");
vertikal.addComponent(button1);
vertikal.addComponent(button2);
vertikal.addComponent(button3);
vertikal.addComponent(button4);
main.addComponent(vertikal);
```

### HorizontalLayout

Kebalikan dari VerticalLayout, HorizontalLayout mengatur letak komponen secara mendatar atau horizontal.

```
HorizontalLayout horison = new HorizontalLayout();
Button button1 = new Button("Button 1");
Button button2 = new Button("Button 2");
Button button3 = new Button("Button 3");
Button button4 = new Button("Button 4");
horison.addComponent(button1);
horison.addComponent(button2);
horison.addComponent(button3);
horison.addComponent(button4);
main.addComponent(horison);
```

## GridLayout

GridLayout mengatur letak komponen berdasarkan grid yang didasarkan oleh nilai baris dan kolom.

```
GridLayout grid = new GridLayout(2,2);
Button button1 = new Button("Button 1");
Button button2 = new Button("Button 2");
Button button3 = new Button("Button 3");
Button button4 = new Button("Button 4");
grid.addComponent(button1);
grid.addComponent(button2);
grid.addComponent(button3);
grid.addComponent(button4);
main.addComponent(grid);
```

## (IV) Database

### Vaadin + MySQL

Untuk dapat mengakses database, seperti biasa kita menggunakan salah satu fitur di Java yaitu JDBC. Sebelum mulai kita buat dulu database dan tabelnya di MySQL. Buka MySQL pada console dengan perintah berikut (pastikan MySQL Server sudah berjalan terlebih dahulu) :

```
kakashi@kakashi:~$ sudo /etc/init.d/mysql start
kakashi@kakashi:~$ mysql -u root -p
Enter password: root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 41
Server version: 5.1.41-3ubuntu12 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.
```

Setelah itu kita akan membuat sebuah database dengan nama **workshop**. Untuk membuatnya ketikkan perintah dibawah ini :

```
mysql> create database workshop;
Query OK, 1 row affected (0.00 sec)
```

Kemudian kita buat tabel-nya dengan nama **mahasiswa**. Ketikkan perintah dibawah ini untuk membuatnya :

```
mysql> use workshop;
Database changed
mysql> create table mahasiswa ( nim varchar(10) not null primary key, nama
```

```
varchar(20), alamat varchar(20) );  
Query OK, 0 rows affected (0.06 sec)
```

Coba kita masukkan beberapa data untuk database dan tabel yang sudah kita buat, caranya seperti dibawah ini :

```
mysql> insert into mahasiswa values ('123090013', 'Joko', 'Merapi');  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into mahasiswa values ('123070073', 'Jono', 'Merbabu');  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into mahasiswa values ('123080063', 'Joni', 'Medan');  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into mahasiswa values ('123060093', 'Johan', 'Malang');  
Query OK, 1 row affected (0.00 sec)
```

Sekarang coba kita panggil data yang sudah kita input dengan perintah dibawah ini :

```
mysql> select * from mahasiswa;  
+-----+-----+-----+  
| nim      | nama  | alamat |  
+-----+-----+-----+  
| 123090013 | Joko  | Merapi  |  
| 123070073 | Jono  | Merbabu |  
| 123080063 | Joni  | Medan   |  
| 123060093 | Johan | Malang  |  
+-----+-----+-----+  
4 rows in set (0.00 sec)
```

Setelah database dan tabel berhasil kita buat, sekarang saatnya untuk membuat aplikasi web-nya menggunakan vaadin. Saya akan memberikan contoh programnya dengan cara yang tidak biasa, saya akan menggunakan konsep DAO Pattern disini. Kenapa menggunakan DAO Pattern? Karena lebih aman, lebih mudah dimaintain dan lebih OOP.

Pertama kali kita buat dulu projectnya, kemudian kita pasang library vaadin & mysql-connector ke dalam project. Buat class dengan nama **Koneksi** dan letakkan didalam package **sample.db.connection**.



```
package sample.db.connection;

import java.sql.DriverManager;
import java.sql.SQLException;
import com.mysql.jdbc.Connection;

public class Koneksi {

    private String url = "";
    private String dbName = "workshop";
    private String host = "localhost";
    private String userName = "root";
    private String password = "root";
    private Connection connection = null;

    public Koneksi() {
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (InstantiationException ex) {
            ex.printStackTrace();
        } catch (IllegalAccessException ex) {
            ex.printStackTrace();
        } catch (ClassNotFoundException ex) {
            ex.printStackTrace();
        }
    }

    public Koneksi(String host, String dbName, String userName, String
password) {
        try {
            this.dbName = dbName;
            this.host = host;
            this.userName = userName;
            this.password = password;
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (InstantiationException ex) {
            ex.printStackTrace();
        } catch (IllegalAccessException ex) {
            ex.printStackTrace();
        }
    }
}
```

```

        } catch (ClassNotFoundException ex) {
            ex.printStackTrace();
        }
    }

    public void closeConnection() {
        try {
            this.connection.close();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    public Connection getConnection() {
        if (this.connection == null) {
            try {
                this.connection = (Connection)
DriverManager.getConnection(getUrl(), getUsername(), getPassword());
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
        return this.connection;
    }

    public void setConnection(Connection connection) {
        this.connection = connection;
    }

    public String getDbName() {
        return dbName;
    }

    public void setDbName(String dbName) {
        this.dbName = dbName;
    }

    public String getHost() {
        return host;
    }

    public void setHost(String host) {
        this.host = host;
    }

```

```

    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
    public String getUrl() {
        String urlNew = "jdbc:mysql://" + getHost() + "/" + getDbName();
        this.url = urlNew;
        return url;
    }
    public void setUrl(String url) {
        this.url = url;
    }
    public String getUserName() {
        return userName;
    }
    public void setUserName(String userName) {
        this.userName = userName;
    }
}

```

Kemudian buat class **Mahasiswa** dan letakkan didalam package **sample.dab.model**.

```

package sample.db.model;

public class Mahasiswa {

    private String nim;
    private String nama;
    private String alamat;

    public String getNim() {
        return nim;
    }
    public void setNim(String nim) {
        this.nim = nim;
    }
    public String getNama() {
        return nama;
    }
    public void setNama(String nama) {
        this.nama = nama;
    }
    public String getAlamat() {

```

```

        return alamat;
    }
    public void setAlamat(String alamat) {
        this.alamat = alamat;
    }
}

```

Buat interface **MahasiswaDao** dan letakkan didalam package **sample.db.dao**.

```

package sample.db.dao;

import java.sql.SQLException;
import java.util.List;
import sample.db.model.Mahasiswa;

public interface MahasiswaDao {
    public void insert(Mahasiswa mahasiswa) throws SQLException;
    public void delete(String nim) throws SQLException;
    public void update(String nim, Mahasiswa mahasiswa) throws
SQLException;
    public List<Mahasiswa> getAll() throws SQLException;
}

```

Buat class **MahasiswaDaoImpl** dan letakkan didalam package **sample.db.dao.impl**.

```

package sample.db.dao.impl;

import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
import com.mysql.jdbc.Connection;
import com.mysql.jdbc.PreparedStatement;
import sample.db.dao.MahasiswaDao;
import sample.db.model.Mahasiswa;

public class MahasiswaDaoImpl implements MahasiswaDao{

    private Connection conn;

    public MahasiswaDaoImpl(Connection conn){
        this.conn = conn;
    }

    @Override
    public void delete(String nim) throws SQLException {

    }

    @Override
    public List<Mahasiswa> getAll() throws SQLException {

```

```

        String sql = "SELECT nim, nama, alamat FROM mahasiswa";
        List<Mahasiswa> list = new ArrayList<Mahasiswa>();
        PreparedStatement st = (PreparedStatement)
conn.prepareStatement(sql);
        ResultSet rs = st.executeQuery();
        while (rs.next()) {
            Mahasiswa mahasiswa = new Mahasiswa();
            mahasiswa.setNim(rs.getString("nim"));
            mahasiswa.setNama(rs.getString("nama"));
            mahasiswa.setAlamat(rs.getString("alamat"));
            list.add(mahasiswa);
        }
        return list;
    }

    @Override
    public void insert(Mahasiswa mahasiswa) throws SQLException {

    }

    @Override
    public void update(String nim, Mahasiswa mahasiswa) throws SQLException
    {

    }

}

```

Sekarang kita buat class dengan nama **TampilkanData** yang berfungsi untuk menampilkan data dan letakkan didalam package **sample.db**.

```

package sample.db;

import java.sql.SQLException;
import java.util.List;
import sample.db.connection.Koneksi;
import sample.db.dao.impl.MahasiswaDaoImpl;
import sample.db.model.Mahasiswa;
import com.mysql.jdbc.Connection;
import com.vaadin.ui.Table;

public class TampilkanData extends Table{

    private static final long serialVersionUID = 1L;

    public TampilkanData(){
        setSizeFull();

        addContainerProperty("Nim", String.class, null);
        addContainerProperty("Nama", String.class, null);
        addContainerProperty("Alamat", String.class, null);

        addData();

        setColumnCollapsingAllowed(true);
        setColumnReorderingAllowed(true);

        setSelectable(true);
    }
}

```

```

        setImmediate(true);

        setNullSelectionAllowed(false);
    }

    public void addData() {
        Koneksi koneksi = new Koneksi();
        Connection connect;
        MahasiswaDaoImpl mahasiswa;

        connect = koneksi.getConnection();
        mahasiswa = new MahasiswaDaoImpl(connect);

        try {
            List<Mahasiswa> list = mahasiswa.getAll();

            for(int i=0; i<list.size(); i++) {
                addItem(new Object[] {list.get(i).getNim(),
list.get(i).getNama(), list.get(i).getAlamat()}, new Integer(i+1));
            }
        } catch (SQLException e) {
            e.printStackTrace();
        } finally {
            koneksi.closeConnection();
        }
    }
}

```

Selanjutnya kita buat class utamanya dengan nama **Main** dan letakkan didalam package **sample.db**.

```

package sample.db;

import com.vaadin.Application;
import com.vaadin.ui.Window;

public class Main extends Application{

    private static final long serialVersionUID = 1L;

    @Override
    public void init() {
        Window window = new Window("Sample Database");
        setMainWindow(window);

        TampilkanData tampil = new TampilkanData();
        window.addComponent(tampil);
    }
}

```

Yang terakhir jangan lupa edit file **web.xml** menjadi seperti berikut :

```

<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://java.sun.com/xml/ns/javaee"

```

```

xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
id="WebApp_ID" version="2.5">

<display-name>WorkshopVaadin</display-name>

<context-param>
  <description>Vaadin production mode</description>
  <param-name>productionMode</param-name>
  <param-value>false</param-value>
</context-param>

<servlet>
  <servlet-name>Workshop Vaadin</servlet-name>
  <servlet-class>
    com.vaadin terminal.gwt.server.ApplicationServlet
  </servlet-class>

  <init-param>
    <description>Vaadin application class to start</description>
    <param-name>application</param-name>
    <param-value>
      sample.db.Main
    </param-value>
  </init-param>
</servlet>

<servlet-mapping>
  <servlet-name>Workshop Vaadin</servlet-name>
  <url-pattern>/*</url-pattern>
</servlet-mapping>

</web-app>

```

Setelah itu baru kita boleh menjalankan programnya.

Untuk proses insert data, ubah class **MahasiswaDaoImpl** tambahkan kode berikut pada method **insert** seperti dibawah ini :

```

@Override
public void insert(Mahasiswa mahasiswa) throws SQLException {
    String sql = "INSERT INTO mahasiswa (nim, nama, alamat) VALUES (?, ?, ?) ";
    PreparedStatement st = (PreparedStatement) conn.prepareStatement(sql);
    st.setString(1, mahasiswa.getNim());
    st.setString(2, mahasiswa.getNama());
    st.setString(3, mahasiswa.getAlamat());
    st.executeUpdate();
}

```

Kemudian buat class baru dengan nama **MasukkanData** dan letakkan didalam package **sample.db**.

```

package sample.db;

import java.sql.SQLException;
import sample.db.connection.Koneksi;

```

```

import sample.db.dao.MahasiswaDao;
import sample.db.dao.impl.MahasiswaDaoImpl;
import sample.db.model.Mahasiswa;
import com.mysql.jdbc.Connection;
import com.vaadin.ui.Button;
import com.vaadin.ui.TextField;
import com.vaadin.ui.VerticalLayout;
import com.vaadin.ui.Button.ClickEvent;
import com.vaadin.ui.Button.ClickListener;

public class MasukkanData extends VerticalLayout{

    private static final long serialVersionUID = 1L;

    public MasukkanData(){
        final TextField nim = new TextField("NIM");
        final TextField nama = new TextField("Nama");
        final TextField alamat = new TextField("Alamat");
        Button ok = new Button("OK");
        addComponent(nim);
        addComponent(nama);
        addComponent(alamat);
        addComponent(ok);

        ok.addListener(new ClickListener() {
            private static final long serialVersionUID = 1L;

            @Override
            public void buttonClick(ClickEvent event) {
                Koneksi koneksi = new Koneksi();
                Connection connect;
                MahasiswaDao mahasiswaDao;

                connect = koneksi.getConnection();
                mahasiswaDao = new MahasiswaDaoImpl(connect);

                Mahasiswa mahasiswa = new Mahasiswa();
                mahasiswa.setNim((String) nim.getValue());
                mahasiswa.setNama((String) nama.getValue());
                mahasiswa.setAlamat((String) alamat.getValue());

                try {
                    mahasiswaDao.insert(mahasiswa);
                } catch (SQLException e) {
                    e.printStackTrace();
                } finally {
                    koneksi.closeConnection();
                }
            }
        });
    }
}

```

Tambahkan kode program dibawah ini ke dalam class **Main**.

```

MasukkanData masuk = new MasukkanData();
window.addComponent(masuk);

```



## (V) Studi Kasus

### Membuat JwebTranslator

Pada studi kasus kali ini kita akan coba membuat aplikasi web untuk penerjemah bahasa, dari bahasa Indonesia ke bahasa Jawa. Seperti biasa kita buat dulu tabel nya seperti dibawah ini.

```
create table bahasa(  
  id int(3) not null primary key auto_increment,  
  indonesia varchar(10),  
  jawahalus varchar(10),  
  jawakasar varchar(10)  
);  
insert into bahasa (indonesia, jawahalus, jawakasar)  
values('saya','kula','aku');  
insert into bahasa (indonesia, jawahalus, jawakasar)  
values('makan','maem','madang');  
insert into bahasa (indonesia, jawahalus, jawakasar)  
values('pergi','tindak','lunga');  
insert into bahasa (indonesia, jawahalus, jawakasar)  
values('sedang','nembe','lagi');
```

Karena programnya terlalu banyak lihat sourcecode nya saja ya, hehehe... :D