

## LAMPIRAN SCRIPT PROGRAM

### ADMIN

#### CSS ADMIN

```
html, body, div, span, applet, object, iframe, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code, del, dfn,
em, font, img, ins, kbd, q, s, samp, small, strike, strong, sub, sup,
tt, var, b, u, i, center, dl, dt, dd, ol, ul, li, fieldset, form, label,
legend, table, caption, tbody, tfoot, thead, tr, th, td
{margin: 0; padding: 0; border: 0; outline: 0; font-size:
100%; vertical-align: baseline;background: transparent;}
body {line-height: 1;}
ol, ul {list-style: none;}
blockquote, q {quotes: none;}
blockquote:before, blockquote:after,
q:before, q:after {content: "";content: none;}
:focus {outline: 0;}
ins {text-decoration: none;}
del {text-decoration: line-through;}
table {border-collapse: collapse;border-spacing: 0;}

.clearfix:after {
content: ".";
display: block;
height: 0;
clear: both;
visibility: hidden; }
.clearfix {display: inline-block;} /* for IE/Mac */
/*-----BASICS-----*/
body{ font-size:12px;font-family: Arial, Helvetica, sans-
serif;color:#000;background:url(..admin.img/bg.jpg) repeat-x
#4780ae;line-height:1.6px;}
a{ color:#162635;text-decoration:none; }
a:hover{text-decoration:underline; }
a:active{ }
a:visited{ }
.left{ float:left; }
.right{ float:right; }
.clear{ clear:both; }
img.left{ margin:0 1em 1em 0; }
img.right{ margin:0 0 1em 1em; }
.padding{ padding:10px; }
/* -----ELEMENTS----- */
h1 {color:#2d2d2d;padding:0px;margin:0px;margin-
top:10px;padding:10px;}
h2 {color:#2d2d2d;font-family: Arial, Helvetica, sans-serif;font-
size:20px;padding:10px;margin:0px;}
h3 {color:#2d2d2d;padding:0px;margin:0px;padding:5px;}
h4 {color:#2d2d2d;padding:0px;margin:0px;padding:5px;}
h5 {color:#2d2d2d;padding:0px;margin:0px;padding:5px;}
p {padding:10px;background:#fff;}
input {padding:3px;border-left:1px solid #a8a8a8;border-top:1px
solid #a8a8a8;border-right:1px solid #d8d8d8;border-bottom:1px
solid #d8d8d8;-moz-border-radius:3px;margin:3px;}
ul {list-style:circle;margin:10px;margin-left:20px;}
ol {list-style:decimal;margin:10px;margin-left:20px;}
p.info {border:1px solid #6aa6d6;background:#c9e3f8;}
fieldset {border:1px solid
#ddd;padding:10px;background:#fff;margin:10px 0px;}
legend {border:1px solid #ddd;padding:5px;background:#fff;}
label {display:block;}
```

```
label input {margin-left:30px;}
hr { border: 1px #ddd solid; width: 80%;}
button {text-transform:uppercase;font-
size:10px;padding:3px;color:#fff;background:#4780ae;border-
left:1px solid #084577;border-top:1px solid #084577;border-
right:1px solid #7ab7e8;border-bottom:1px solid #7ab7e8;-moz-
border-radius:3px;margin:3px;margin-top:5px;}
/* -----Login----- */
#login {background:url(..admin.img/security.jpg) #fff 220px 55px
no-repeat;padding-left:10px;text-align:left;}
#login_container { overflow:hidden; width:360px; margin:0
auto;background:url(..admin.img/overlay.jpg) no-repeat;}
#login label span {display:block;width:100px;float:left}
div#login {width:260px;margin: 10px auto;border-top:2px solid
#162635;}
a#passwordrecoverylink {font-size:10px;text-
decoration:underline;display:block;text-align:right;color:#999;}
a:hover#passwordrecoverylink {color:#000;}
.loginbutton {margin-left:160px;}
#login input#remember {margin:10px 5px 0px 10px;}
/* ----- MENUS ----- */
#navigation, #navigation ul { padding:0; margin:0; list-style:none;
line-height:1; float:left; }
#navigation a { display:block; padding:6px 10px; text-
decoration:none; }
#navigation a:hover { text-decoration:underline; padding:6px
10px;}
#navigation li { float:left; width:auto; }
#navigation li { padding:0; margin:0; }
#navigation li ul li { padding:1px 0px;}
#navigation li ul li a { padding:6px 10px; width:110px; }
#navigation li ul { position:absolute; padding-top:1px;
width:130px; left:-999em; }
#navigation li ul ul { margin: -1.5em 0 0 130px; }
#navigation li: hover ul ul, #navigation li: hover ul ul ul,
#navigation li.sfhover ul ul, #navigation li.sfhover ul ul ul { left: -
999em;}
#navigation li: hover ul, #navigation li li: hover ul, #navigation li li
li: hover ul,
#navigation li.sfhover ul, #navigation li li.sfhover ul, #navigation li
li li.sfhover ul { left: auto; }
#top_menu ul { margin:0px;padding:0px;list-style:none;font-
size:12px;}
#top_menu ul li {float:left;display:block;}
#top_menu ul li a {text-
decoration:none;color:#fff;display:block;padding:12px 15px 10px
15px;}
#top_menu ul li
a:hover {background:url(..admin.img/menutop_hover.jpg) bottom
center no-repeat}
/* ----- STRUCTURE ----- */
#container { overflow:hidden; width:960px; margin:0
auto;background:url(..admin.img/overlay.jpg) no-repeat;}
#header { }
#footer {clear:both; }
#content_main {background:#f1f1f1;width:940px;padding:10px;-
moz-border-radius:3px;}
#top_menu {background:url(..admin.img/menutop.jpg) no-repeat
top center;width:960px;margin-top:30px;height:35px;}
```

```

#profile_info{ background:url(..admin.img/loginbox.jpg) no-repeat top
center;width:231px;height:59px;padding:4px;float:right;margin-top:35px;font-size:12px;color:#fff;line-height:10px;}
#profile_info p{background:none;}
#avatar{float:left;margin:0px 10px;}
#postedit{ background:#f1f1f1;width:940px;padding:10px;-moz-border-radius:3px;}
#dashboard{ width:640px;background:#fff;padding:10px;border-top:1px solid #ddd;border-bottom:1px solid #ddd;}
#sidebar{width:250px;background:#fff;padding:10px;border-top:1px solid #ddd;border-bottom:1px solid #ddd;-moz-border-radius:3px;}
#footer{background:#f1f1f1;width:940px;padding:10px;-moz-border-radius:3px;margin-top:10px;}
#footer p{background:none;padding:0px;margin:0px}
.panel{ min-height:285px;}
.section,.panel,#shortcuts,#chart,.quickview{ background:#fff;padding:10px;border-top:1px solid #ddd;border-bottom:1px solid #ddd;-moz-border-radius:3px;}
#postedit,#panels,#shortcuts{margin-top:10px;}
.section{margin-top:10px;}
.panel {width:290px;margin:5px;}
.calendar {width:210px;margin-right:0px;}
.photo {width:375px;margin-left:0px;margin-right:10px;}
.form_sidebar{width:220px;margin-left:10px;margin-top:5px;padding:4px;background:#fff;border-top:1px solid #ddd;border-bottom:1px solid #ddd;}
#form_middle_cont{width:940px}
/* ----- ICONS ----- */
.ico_mug{background:url(..admin.img/ico_mug.jpg) no-repeat 5px center #f1f1f1; padding-left:40px;margin-bottom:20px;}
.ico_posts{background:url(..admin.img/ico_posts.jpg) no-repeat 5px center #f1f1f1; padding-left:30px;}
.ico_page{background:url(..admin.img/ico_page.jpg) no-repeat 5px center #f1f1f1; padding-left:30px;}
.ico_settings{background:url(..admin.img/ico_settings.jpg) no-repeat 5px center #f1f1f1; padding-left:30px;}
.ico_user{background:url(..admin.img/ico_user.jpg) no-repeat 5px center #f1f1f1; padding-left:30px;}
.ico_success{padding-left:20px;background:url(..admin.img/success.jpg) no-repeat left center}
.ico_cancel{padding-left:20px;background:url(..admin.img/error.jpg) no-repeat left center}
.ico_error{padding-left:20px;background:url(..admin.img/ico_error.jpg) no-repeat left center}
.ico_pending{padding-left:20px;background:url(..admin.img/ico_hourglass.jpg) no-repeat left center}
.ico_chart{padding-left:20px;background:url(..admin.img/ico_chart_bar.jpg) no-repeat left center}
/* ----- LOGO ----- */
div#logo h1 {padding:0px;margin:0px;}
div#logo h1 a{display:block;text-indent:-9999px;width:367px;height:107px;background:url(..admin.img/logo.jpg) no-repeat;}
/* -----CUSTOM -----*/

#placeholder{ width:180px;height:95px;padding:10px;padding-bottom:14px;}
.more{ font-size:13px;}
.approved{ font-weight:bold;color:#25a21f}
#success{ margin:-px 0px;border:1px solid #b2dc4d;color:#40550d;font-family:"Arial", Arial, sans-serif;font-size:12px;background:#cce297;font-weight:700;padding:5px;-moz-border-radius:3px;-webkit-border-radius: 3px;}
#fail{margin:5px 0px;border:1px solid #c82820;color:#c82820;font-family:"Arial", Arial, sans-serif;font-size:12px;font-weight:700;background:#e7928d;padding:5px;-moz-border-radius:3px;-webkit-border-radius: 3px;}
#warning{margin:5px 0px;border:1px solid #efdc90;color:#a9a014;font-family:"Arial", Arial, sans-serif;font-size:12px;font-weight:700;background:#fffecf;padding:5px;-moz-border-radius:3px;-webkit-border-radius: 3px;}
.form_sidebar ul{ margin:0px;padding:0px;list-style:none;}
.form_sidebar ul li ul{margin:0px 10px;padding:0px;list-style:none;}
span.date{display:block;padding-left:20px; font-style:italic;font-family:10px;}
.odd{background:#eee;border-top:1px solid #DDDDDD;padding:3px;}
.even{ border-top:1px solid #DDDDDD;padding:3px;}
a#visit{ display:block;color:#fff;background:#4780ae;border-left:1px solid #084577;border:1px solid #7ab7e8;padding:4px;margin:5px 15px 0px 0px;-moz-border-radius:3px;}
a.add{display:block;width:80px;text-align:center;color:#fff;background:#4780ae;border:1px solid #4780ae;padding:4px;margin:10px 15px 0px 0px;-moz-border-radius:3px;text-transform:uppercase;}
#profile_info p{font-size:11px;padding-top:2px;padding-bottom:4px;}
#profile_info p a{font-size:11px;color:#f9d71c;}
#profile_info p a: hover{font-size:11px;color:#f7d202;text-decoration:underline;}
#profile_info p.last_login{font-size:10px;padding-top:5px;color:#dfdfdf}
#table_options ul li a {color:#000;display:block;padding-top:5px;}
.quickview ul{margin:0px;padding:0px;list-style:none;}
.quickview ul li{padding:3px;font-size:12px;}
.quickview ul li span.number{font-family:Georgia;font-size:13px;color:#629fa8}
#shortcuts{width:640px;}
#shortcuts ul {margin:0px;padding:0px;list-style:none;margin-top:20px;}
#shortcuts ul li{float:left}
#shortcuts ul li.first_li a{margin-left:0px;}
#shortcuts ul li a{border:1px solid #999;-moz-border-radius:3px;display:block;padding:5px;margin:5px;width:70px;text-align:center;}
#shortcuts ul li a span{font-size:10px;margin-top:5px;display:block;background:#eee;padding:2px;}
#shortcuts ul li a: hover{border:1px solid #ececce;text-decoration:none;}
.photo ul{margin:0px;padding:0px;list-style:none;}
.photo ul li{float:left; border:1px solid #999;-moz-border-radius:3px;padding:3px;margin:3px 1px;position:relative;}

```

```

.photo ul li span{ font-size:10px;margin-
top:5px;display:block;padding:4px
1px;width:83px;position:absolute;bottom:0px;left:0px;}
.todo ul {margin:0px;padding:0px;list-style:none;}
.todo ul li {margin-left:0px;padding-left:0px;list-style:none;}
.todo p {margin:0px;padding:0px;}
ul#menu {margin:0px;padding:0px;list-style:none;margin-
top:20px;}
ul#menu, ul#menu ul {list-style-type:none; margin: 0; padding: 0;
}
ul#menu a { display: block; text-decoration: none;
}
ul#menu li { margin-top: 1px;}
ul#menu li a { color: #000;padding: 10px;padding-
left:25px;border-bottom:1px solid #ddd}
ul#menu li a:hover { }
ul#menu li ul li a { background: #fff; color: #000; padding-left:
15px; border-bottom:none}
ul#menu li ul li a:hover { background: #eee; border-left: 5px
#4780ae solid; padding-left: 10px;}
input#post_title{width:925px;margin:0px;border-left:1px solid
#a8a8a8;border-top:1px solid #a8a8a8;border-right:1px solid
#d8d8d8;border-bottom:1px solid #d8d8d8;-moz-border-
radius:3px;font-size:28px;padding:5px;}
.form_sidebar span{display:block;padding:5px
5px;background:#eee;margin-bottom:10px;}
#save{ width:100px;color:#fff;background:#4780ae;border-left:1px
solid #084577;border-top:1px solid #084577;border-right:1px solid
#7ab7e8;border-bottom:1px solid #7ab7e8;}
#table{ width:100%;border:1px solid #c3c3c3;}
#table th{background:#eee;padding:5px;text-align:left;}
#table td{padding:5px;}
#table .table_date{color:#5e5e5e;width:100px;font-weight:bold;}
#table .table_check{ width:20px;background:#eee;}
#table .table_title a{ color:#26608e;text-decoration:underline;font-
weight:bold;}
#table_options{background:#fff;border:solid #c3c3c3;border-
width:0px 1px 1px 1px;padding:5px;}
#table_options label{margin:0px;padding-top:2px;}
#table_options ul{list-style:none;margin:0px;padding:0px;}
#table_options ul li{float:left;padding:5px;margin-right:10px;}
.pagination {margin-top:20px;}
.pagination a, .pagination
span{background:url("../admin.img/borderbottom.jpg") bottom
repeat-x #fff;padding:5px;margin:2px;border:1px solid
#d9d9d9;text-decoration:none;-moz-border-radius:3px;-webkit-
border-radius: 3px;}
.file{
border:1px solid #eee;
}

```

#### APPLICATION.JAVA

```

package com.roes.commerce.pages;
public class Application extends WebApplication{
@Override
public Class<? extends Page> getHomePage() {
return Index.class; }
@Override
public Session newSession(Request req, Response res){
return new WicketSession(req); } }

```

#### HOMEPAGE.JAVA

```

package com.roes.commerce.pages;
import java.io.ByteArrayOutputStream;
public class HomePage extends WebPage {
private ByteArrayOutputStream outputStream;
public HomePage(PageParameters parametersPage) {
//validate();
User user = ((WicketSession) getSession()).getUser();
try {
if(user.equals(null))
setResponsePage(Login.class);
} catch (Exception e) {
setResponsePage(Login.class);
e.printStackTrace(); }
BookmarkablePageLink lnkUserPage = new
BookmarkablePageLink(
"InkUserPage", UserViewPage.class) { };
BookmarkablePageLink category = new
BookmarkablePageLink("category",
CategoryViewPage.class) { };
BookmarkablePageLink product = new
BookmarkablePageLink("product",
ProductViewPage.class) { };
BookmarkablePageLink manufacture = new
BookmarkablePageLink("manufacture",
ManufactureViewPage.class) { };
Link logout = new Link("logout") {
@Override
public void onClick() {
((WicketSession) getSession()).setUser(null);
getSession().invalidate();
setResponsePage(Login.class); } };
add(new Label("username", new Model(user.getUsername())));
add(new Label("privilege", new
Model(user.getUserGroup().getName()));
add(logout);
add(lnkUserPage);
add(category);
add(product);
add(manufacture);
add(new PageLink("lnkCustomerPage",
CustomerPage.class));
add(new PageLink("dashboard",
Dashboard.class));
add(new PageLink("orders",
OrderViewPage.class)); }
private void validate() {
try {
User user = ((WicketSession) getSession()).getUser();
System.out.println("data "+user);
System.out.println("data2 "+user);
if (user == null) {
System.out.println("data3 "+user);
setResponsePage(Login.class); }
} catch (Exception e) {
e.printStackTrace(); } }
public void addCssPage(String source) {
add(HeaderContributor.forCss(source)); }
public void addJsPage(String source) {
add(HeaderContributor.forJavaScript(source)); }
public void logout() {

```

```

        getSession().invalidate();
        setResponsePage(Login.class); }
public void giveAlert(AjaxRequestTarget target, String msg) {
    target.appendJavaScript("jAlert('info','" + msg + "','Info
Dialog');"); }
public void giveJAlert(AjaxRequestTarget target, String msg) {
    target.appendJavaScript("jAlert('error,'" + msg
+ "','Error Dialog');"); }
public TextField createTextField(String id, boolean isMarkUp,
boolean isRequired) {
    TextField textfield = (TextField) new TextField(id, new
Model())
        .setOutputMarkupId(isMarkUp);
    textfield.setRequired(isRequired);
    return textfield; }
public DropDownChoice createDropDownChoice(String id,
boolean isMarkUp,
boolean isRequired, List listData) {
    DropDownChoice dc = new DropDownChoice(id, new Model(),
listData);
    dc.setOutputMarkupId(isMarkUp);
    dc.setRequired(isRequired);
    return dc; }
public FeedbackPanel createFeedBackPanel(String id, boolean
isMarkUp) {
    FeedbackPanel feedback = (FeedbackPanel) new
FeedbackPanel(id)
        .setOutputMarkupId(true);
    return feedback; }
public void addToForm(Form form, Component[] components) {
    for (Component component : components) {
        form.add(component); } }
public void saveAjaxProcessing(AjaxRequestTarget target, String
msg,
    FeedbackPanel feedback) {
    target.addComponent(feedback);
    giveAlert(target, msg); }
public void refreshComponents(AjaxRequestTarget target,
Component[] components) {
    for (Component component : components) {
        target.addComponent(component); } }
public void setOutputMarkupIdComponent(Component[]
components,
boolean output) {
    for (Component component : components) {
        component.setOutputMarkupId(output); } }
public void setObjectComponent(Component component, Object
object) {
    component.getModel().setObject(object); }
public void addLabelToItem(Label[] label, ListItem item) {
    for (Label label2 : label) {
        item.add(label2); } }

```

## HOMEPAGE.HTML

```

<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>AdminTheme - IT Shop. Roes Corp</title>
<meta name="description" content="" />
<meta name="keywords" content="" />
<meta name="robots" content="index,follow" />
<link rel="stylesheet" type="text/css" media="all"

```

```

href="admin.css/style.css" />
<link rel="stylesheet" type="text/css"
href="admin.css/superfish.css" />
<script type="text/javascript" src="admin.js/jquery-
1.3.2.min.js"></script></head>
<body>
    <div class="container" id="container">
    <div id="header">
    <div id="profile_info">

    <p>
    <label>Welcome
    <strong><span wicket:id="username">[UserName]
    </span> As <span wicket:id="privilege">
    [UserName]</span></strong>.
    <a wicket:id="logout">Log
out?</a></label></p></div>
    <div id="logo"> </div> </div>
    <!-- end header -->
    <div id="content">
    <div id="top_menu" class="clearfix">
    <ul class="sf-menu">
    <!-- DROPDOWN MENU -->
    <li><a href="#" wicket:id="dashboard">BERANDA</a></li>
    <li><a href="#" wicket:id="product">PRODUK</a></li>
    <li><a href="#" wicket:id="category">KATEGORI</a></li>
    <li><a wicket:id="InkUserPage">USER</a></li>
    <li><a href="#"
wicket:id="InkCustomerPage">KONSUMEN</a></li>
    <li><a href="#"
wicket:id="manufacture">MANUFAKTUR</a></li>
    <li><a href="#" wicket:id="orders">PEMESANAN</a></li></ul>
    </div>
    <wicket:child></wicket:child>
    <div id="footer" class="clearfix">
    <p class="left">AdminTheme - IT Shop. Roes Corp</p>
    <p class="right">
    © 2011 radhitheme </p> </div>

```

## LOGIN.JAVA

```

package com.roes.commerce.pages;
public class Login extends WebPage {
    private ServiceCRUDUser serviceCRUDUser =
ServiceCRUDUser.getInstance();
    public Login() {
        FeedbackPanel msg = new
FeedbackPanel("msg");
        final TextField name = (TextField) new
TextField("name", new Model()).setRequired(true);
        final PasswordTextField password =
(PasswordTextField) new PasswordTextField("password", new
Model()).setRequired(true);
        Form form = new Form("form");
        Button button = new Button("btnLogin");
    public void onSubmit() { User user =
serviceCRUDUser.getUserByNameAndPass(name.getValue(),
password.getValue());
        if(user==null){
            setResponsePage(Login.class);
        }else{
            ((WicketSession)getSession()).setUser(user);
            setResponsePage(Dashboard.class); } } };
        form.add(name);
        form.add(password);
        form.add(button);
        add(msg);

```

```
add(form); } }
```

## LOGIN.HTML

```
<html>
<head>
  <title>Login.html</title>
  <meta http-equiv="keywords"
content="keyword1,keyword2,keyword3">
  <meta http-equiv="description" content="this is my page">
  <meta http-equiv="content-type" content="text/html;
charset=ISO-8859-1"> </head>
<body>
  <span wicket:id="msg"></span>
  <form name="f1" id="f1" action="" method="post"
wicket:id="form">
    <table border="0">
      <tr><td>Login:</td>
        <td><input type="text" wicket:id="name"></td></tr>
      <tr><td>Password:</td>
        <td><input type="password"
wicket:id="password"></td></tr>
      <tr><td colspan="2" align="center"><input type="submit"
wicket:id="btnLogin"></td></tr> </table></form></body>
</html>-->
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-
8" />
<title>AdminTheme - Ultimate Admin Panel Solution</title>
<link rel="stylesheet" type="text/css" media="all"
href="admin.css/style.css" />
<link rel="stylesheet" type="text/css"
href="admin.css/smoothness/jquery-ui-1.7.1.custom.css" />
<script type="text/javascript" src="admin.js/jquery-
1.3.2.min.js"></script>
<script type="text/javascript" src="admin.js/jquery-ui-
1.7.1.custom.min.js"></script>
<script type="text/javascript" src="admin.js/custom.js"></script>
</head>
<body>
<div id="login_container">
<div id="header">
<div id="logo">
<h1><a href="/">AdmintTheme</a></h1></div></div>
<!-- end header -->
<div id="login" class="section">
<div id="fail" class="info_div">
<span class="ico_cancel">MASUKKAN NAMA ADMIN dan
PASSWORD</span> </div> <span wicket:id="msg"></span>
<form wicket:id="form" name="loginform" id="loginform"
action="index.html"
method="post">
  <label>
    <strong>NAMA ADMIN</strong></label>
  <input wicket:id="name" type="text" name="log" id="user_login"
size="28" class="input" /> <br />
  <label>
    <strong>PASSWORD</strong></label>
  <input wicket:id="password" type="password" name="pwd"
id="user_pass" size="28" class="input" /> <br />
    <strong>Remember Me</strong>
  <input type="checkbox" id="remember" class="input noborder" />
    <br />
  <input wicket:id="btnLogin" id="save" class="loginbutton"
type="submit"
value="Log In" /></form>
<a href="#" id="passwordrecoverylink">Forgot your username or
password?</a></div></div>
<!-- end container -->
</body></html>
```

## CATEGORYPAGE.JAVA

```
package com.roes.commerce.pages.admin;
import java.util.Date;
public class CategoryPage extends HomePage {
private ServiceCRUDCategory serviceCRUDCategory =
ServiceCRUDCategory.getInstance();
public CategoryPage(PageParameters parametersPage)
{ super(parametersPage);
FeedbackPanel msg = new FeedbackPanel("msg");
Form form = new Form("form");
final TextField name = createTextField("name", false, true);
final TextArea desc = (TextArea) new TextArea("desc", new
Model()).setRequired(true);
final DropDownChoice parent = createDropDownChoice("parent",
false,false, serviceCRUDCategory.getAllNameCategory());
Button btnSave = new Button("btnSave") {
public void onSubmit() {
Category category = new Category();
category.setDateAdded(new Date());
category.setDateModified(new Date());
if (!parent.getValue().equals("-1")) {
Integer numb = Integer.parseInt(parent.getValue());
String nameDesc = serviceCRUDCategory.getAllNameCategory()
.get(numb);
CategoryDescription categoryDescription =serviceCRUDCategory
.getCategoryDescByName(nameDesc);
category.setParentId(categoryDescription.getCategory()); }
CategoryDescription categoryDescription = new
CategoryDescription();
categoryDescription.setDescription(desc.getValue());
categoryDescription.setName(name.getValue());
categoryDescription.setCategory(category);
serviceCRUDCategory.saveCategory(category,
categoryDescription); } };
Component[] components = { name, desc, parent, btnSave };
addToForm(form, components);
add(form);
add(msg); } }
```

## CATEGORYPAGE.HTML

```
<wicket:extend>
<div id="tabledata" class="section">
<h1>Form Input Category</h1>
<span wicket:id="msg"></span>
<form wicket:id="form" >
<table border="0">
<tr><td>NAMA:</td>
  <td><input type="text" wicket:id="name"></td></tr>
<tr><td>KETERANGAN:</td>
  <td><textarea wicket:id="desc"></textarea></td></tr>
<tr><td>Induk:</td>
  <td><select wicket:id="parent"></td></tr>
<tr>
  <td colspan="2" align="center"><input type="submit"
value="Save" wicket:id="btnSave"></td>
  <td colspan="2" align="center"><input type="reset"
value="Reset"></td>
</tr>
</table>
</form>
```

```

</div>
</wicket:extend>

```

### CATEGORYVIEWPAGE.JAVA

```

package com.roes.commerce.pages.admin;
import java.util.List;
public class CategoryViewPage extends HomePage {
    private ServiceCRUDCategory serviceCRUDCategory
= ServiceCRUDCategory.getInstance();
    public CategoryViewPage(PageParameters
parametersPage) { super(parametersPage);
    PageableListView dataView =
createTableData("categories",
serviceCRUDCategory.getAllCategory(), 10);
BookmarkablePageLink linknew = new
BookmarkablePageLink("linknew",
CategoryPage.class,new PageParameters());
        add(dataView);
        add(linknew); }
    private PageableListView createTableData(String id,
List list,
Integer paging) {
PageableListView dataView = new PageableListView(id, list,
paging) {
@Override
protected void populateItem(ListItem item) {
final Category category = (Category) item.getModelObject();
item.setModel(new CompoundPropertyModel(category));
Category parentId = category.getParentId();
if(parentId==null){
    item.add(new Label("parent", new Model(""));
    }else{
    item.add(new Label("parent", new
Model(parentId.getCategoryDescription().getName())); }
    item.add(new Label("name", new Model(category
.getCategoryDescription().getName()));
    item.add(new Label("description", new Model(category
.getCategoryDescription().getDescription()));
    AjaxFallbackLink lnkdelete = new
AjaxFallbackLink("lnkdelete") {
@Override
public void onClick(AjaxRequestTarget target) {
try {
serviceCRUDCategory.deleteCategory(category);
setResponsePage(CategoryViewPage.class);
} catch (Exception e) {
e.printStackTrace(); } } };
        item.add(lnkdelete);
    } }; return dataView; } }

```

### CATEGORYVIEWPAGE.HTML

```

<wicket:extend>
<div id="tabledata" class="section">
<h2 class="ico_mug">DAFTAR KATEGORI</h2><a
wicket:id="linknew" style="float: left; clear: both;">BUAT
BARU</a>
<table id="table">
<thead>
<tr>

```

```

<th>INDUK</th>
<th>NAMA</th>
<th>KETERANGAN</th>
<th>TINDAKAN</th></tr></thead>
<tbody>
<tr wicket:id="categories">
<td class="table_title" wicket:id="parent"></td>
<td class="table_title" wicket:id="name"></td>
<td class="table_title" wicket:id="description"></td>
<td>
<a href="#" wicket:id="lnkdelete" title="delete this book"
class="vtip">delete</a>
</td></tr></tbody></table></div>
</wicket:extend>

```

### CUSTOMERPAGE.JAVA

```

package com.roes.commerce.pages.admin;
import java.util.List;
public class CustomerPage extends HomePage {
    private ServiceCRUDCustomer serviceCRUDCustomer
= ServiceCRUDCustomer.getInstance();
    public CustomerPage(PageParameters parametersPage)
{super(parametersPage);
PageableListView dataView = createTableData("users",
serviceCRUDCustomer.getAllWaiting(), 10);
    PagingNavigator paginator = new
PagingNavigator("paginator", dataView);
        add(dataView);
        add(paginator); }
    private PageableListView createTableData(String id,
List list, Integer paging) {
PageableListView dataView = new PageableListView(id, list,
paging) {
@Override
protected void populateItem(ListItem item) {
final Customer customer = (Customer) item.getModelObject();
item.setModel(new CompoundPropertyModel(customer));
item.add(new Label("telephone", new
Model(customer.getTelephone()));
item.add(new Label("firstname", new
Model(customer.getFirstname()));
item.add(new Label("lastname", new
Model(customer.getLastname()));
item.add(new Label("email", new Model(customer.getEmail()));
item.add(new Label("address", new
Model(customer.getAddress().getAddressI()));
item.add(new Label("city", new
Model(customer.getAddress().getCity()+"["+customer.getAddress(
).getPostcode()+"]"));
    AjaxFallbackLink lnkdelete = new
AjaxFallbackLink("lnkdelete") {
@Override
public void onClick(AjaxRequestTarget target) {
try {
serviceCRUDCustomer.deleteCustomer(customer);
setResponsePage(CustomerPage.class);
} catch (Exception e) {
e.printStackTrace(); } } };
        item.add(lnkdelete); } };
    return dataView; } }

```

## CUSTOMERPAGE.HTML

```
<wicket:extend>
<div id="tabledata" class="section">
<h2 class="ico_mug">DAFTAR KONSUMEN</h2>
<table id="table">
<thead><tr>
<th>NAMA</th>
<th>NAMA LENGKAP</th>
<th>EMAIL</th>
<th>TELEPON</th>
<th>ALAMAT</th>
<th>KOTA [KODE POS]</th>
<th>TINDAKAN</th></tr></thead>
<tbody>
<tr wicket:id="users">
<td class="table_title" wicket:id="firstname"></td>
<td class="table_title" wicket:id="lastname"></td>
<td class="table_date" wicket:id="telephone"></td>
<td class="table_title" wicket:id="email"></td>
<td class="table_title" wicket:id="address"></td>
<td class="table_title" wicket:id="city"></td>
<td>
<a href="#" wicket:id="lnkdelete" title="delete this
book" class="vtip">delete</a> </td> </tr> </tbody> </table>
<div class="pagination" style="float: left; width: 800px;">
<span wicket:id="paginator" /></div></div>
</wicket:extend>
```

## DASHBOARD.JAVA

```
package com.roes.commerce.pages.admin;
import org.apache.wicket.PageParameters;
import com.roes.commerce.pages.HomePage;
public class Dashboard extends HomePage{
    public Dashboard(PageParameters parameters){
        super(parameters);
    }
}
```

## DASHBOARD.HTML

```
<wicket:extend>
<div id="content_main" class="clearfix">
<div id="main_panel_container" class="left">
<div id="dashboard">
<h2 class="ico_mug">BERANDA</h2>
<div class="clearfix">
<div id="chart" class="left">
<h3>SILAHKAN PILIH ITEM ADMIN
SEPERLUNYA</h3>
<div id="placeholder" ></div><!-- CHART -->
</div> </div> </div><!-- end #dashboard -->
</wicket:extend>
```

## MANUFACTURPAGE.JAVA

```
package com.roes.commerce.pages.admin;
import java.util.Date;
public class ManufacturePage extends HomePage {
    private ServiceCRUDProduct serviceCRUDProduct =
ServiceCRUDProduct.getInstance();
    private TextField name;
    private Form form;
    private FeedbackPanel msg;
    boolean isEdit = false;
    private Manufacture manufactureEdit;
```

```
    public ManufacturePage(PageParameters
parametersPage) { super(parametersPage);
    name = createTextField("name", false, true);
    if(parametersPage.getString("page")!=null){
    isEdit = true;
    if(parametersPage.getString("page").equals("edit")){
    Integer id =
Integer.parseInt(parametersPage.getString("id"));
    manufactureEdit = serviceCRUDProduct.getManufactureById(id);
    name.setModelObject(manufactureEdit.getName()); } }
    form = new Form("form") {
    public void onSubmit() {
    try {
    if(isEdit){
    manufactureEdit.setName(name.getValue())
    serviceCRUDProduct.saveManufacture(manufactureEd
it);
    }else{
    Manufacture manufacture = new Manufacture();
    manufacture.setName(name.getValue());
    serviceCRUDProduct.saveManufacture(manufacture);}
    } catch (Exception e) {
    e.printStackTrace(); } } };
    msg = new FeedbackPanel("msg");
    Component[] components = { name };
    addToForm(form, components);
    add(form);
    add(msg); } }
```

## MANUFACTURPAGE.HTML

```
<wicket:extend>
<div id="tabledata" class="section">
<span wicket:id="msg" style="float: left; clear: both;"></span>
<form wicket:id="form">
<table border="0">
<tr><td>NAMA:</td>
<td><input type="text" wicket:id="name"/></td></tr>
<tr><td colspan="2" align="center"><input type="submit"
value="Save"> </td></tr> </table> </form> </div>
</wicket:extend>
```

## MANUFACTURVIEWPAGE.JAVA

```
package com.roes.commerce.pages.admin;
import java.util.List;
public class ManufactureViewPage extends HomePage {
    private ServiceCRUDProduct serviceCRUDProduct =
ServiceCRUDProduct.getInstance();
    public ManufactureViewPage(PageParameters
parametersPage) { super(parametersPage);
    List<Manufacture> all =
serviceCRUDProduct.getAllManufacture();
    PageableView dataView =
createTableData("products", all, 10);
    BookmarkablePageLink linknew = new
BookmarkablePageLink("linknew",
ManufacturePage.class, new PageParameters());
    PagingNavigator paginator = new PagingNavigator("paginator",
dataView);
    add(dataView);
    add(linknew);
    add(paginator); }
```

```

        private PageableView createTableData(String
string, List<Manufacture> all, int i) {
        PageableView listView = new
PageableView(string, all, i) {
        int x = 1;
@Override
protected void populateItem(ListItem item) {
try {
final Manufacture manufactur = (Manufacture) item
.getModelObject();
final Label no = new Label("no", new Model(x++));
final Label name = new Label("name", new
Model(manufactur.getName()));
AjaxFallbackLink lnkdelete = new AjaxFallbackLink(
"lnkdelete") {
@Override
public void onClick(AjaxRequestTarget target) {
try {
serviceCRUDProduct
.deleteManufacture(manufactur);
setResponsePage(ManufactureViewPage.class); } catch
(Exception e) {
e.printStackTrace(); } } };
AjaxFallbackLink lnkedit = new
AjaxFallbackLink("lnkedit") {
@Override
public void onClick(AjaxRequestTarget target) {
PageParameters parameters = new PageParameters();
parameters.add("page", "edit");
parameters.add("id", manufactur.getManufacturerId()
.toString());
setResponsePage(ManufacturePage.class, parameters);
} };
item.add(no);
item.add(name);
item.add(lnkdelete);
item.add(lnkedit);
} catch (Exception e) {
e.printStackTrace(); } } };
return listView; } }

```

#### MANUFACTUREVIEWPAGE.HTML

```

<wicket:extend>
<div id="tabledata" class="section">
<h2 class="ico_mug">DAFTAR MANUFAKTUR</h2>
<a wicket:id="linknew" style="float: left; clear: both;">LINK
NEW</a>
<table id="table">
<thead>
<tr><th>NO</th>
<th>NAMA</th>
<th>TINDAKAN</th></tr></thead>
<tbody>
<tr wicket:id="products">
<td class="table_check" wicket:id="no"></td>
<td wicket:id="name"></td>
<td><a href="#" wicket:id="lnkdelete" title="delete this
MANUFACTUR" class="vtip">HAPUS</a>
<a href="#" wicket:id="lnkedit" title="edit this book"
class="vtip">edit</a> </td> </tr> </tbody> </table>
<div class="pagination" style="float: left; width: 800px;">

```

```

<span wicket:id="paginator" /> </div> </div>
</wicket:extend>

```

#### ORDERVIEWPAGE.JAVA

```

package com.roes.commerce.pages.admin;

import java.util.List;
import org.apache.wicket.PageParameters;
import org.apache.wicket.ajax.AjaxRequestTarget;
import org.apache.wicket.ajax.markup.html.AjaxFallbackLink;
import org.apache.wicket.markup.html.basic.Label;
import org.apache.wicket.markup.html.list.ListItem;
import org.apache.wicket.markup.html.list.PageableView;
import org.apache.wicket.markup.html.navigation.paging.PagingNavigator;
import org.apache.wicket.model.CompoundPropertyModel;
import org.apache.wicket.model.Model;
import com.roes.commerce.model.Customer;
import com.roes.commerce.model.OrderStatus;
import com.roes.commerce.model.Orders;
import com.roes.commerce.model.OrdersProduct;
import com.roes.commerce.model.Product;
import com.roes.commerce.pages.HomePage;
import com.roes.commerce.service.ServiceCRUDOrder;
import com.roes.commerce.tools.Constants;
import com.roes.commerce.tools.SendMail;
public class OrderViewPage extends HomePage {
private ServiceCRUDOrder serviceCRUDOrder =
ServiceCRUDOrder.getInstance();
public OrderViewPage(PageParameters
parametersPage) {
super(parametersPage);
PageableView dataView =
createTableData("users", serviceCRUDOrder.getOrdersWaiting(),
10);
PagingNavigator paginator = new
PagingNavigator("paginator", dataView);
add(dataView);
add(paginator); }
private PageableView createTableData(String id,
List<Orders> list,
Integer paging) {
PageableView dataView = new
PageableView(id, list, paging) {
@Override
protected void populateItem(ListItem item) {
final Orders orders = (Orders) item.getModelObject();
final Customer customer = orders.getCustomer();
item.setModel(new
CompoundPropertyModel(orders);
item.add(new Label("orderid", new
Model(orders.getOrderid()));
item.add(new Label("customer", new
Model(orders.getCustomer().getFirstname())));
int total = 0;
final List<OrdersProduct>
orderProductByOrderId =
serviceCRUDOrder.getOrderProductByOrderId(orders.getOrderid(
));

```



```

        if(orderProductByOrderId.size()>0){
            total =orderProductByOrderId.size(); }
            item.add(new Label("total", new
Model(total)));
            item.add(new Label("status", new
Model(orders.getOrderStatus().getName()));
String nameAction = "";
                final Integer
orderStatusId = orders.getOrderStatus().getOrderStatusId();
                if(orderStatusId.equals(new Integer(14))){
                    nameAction = "Approve";
                }else{
                    nameAction = "Shipping"; }
AjaxFallbackLink lnkDetail = new AjaxFallbackLink("lnkDetail")
{
@Override
    public void onClick(AjaxRequestTarget target) {
        PageParameters parameters = new PageParameters();
        parameters.add("orderId",
orders.getId().toString());
        setResponsePage(OrderDetailViewPage.class,
parameters); } };
        item.add(lnkDetail);
        Label statusaction = new Label("statusaction", new
Model(nameAction));
        AjaxFallbackLink action = new
AjaxFallbackLink("action"){
@Override
    public void onClick(AjaxRequestTarget target) {
        if(orderStatusId.equals(new Integer(14))){
//update status orders
            OrderStatus orderStatus =
serviceCRUDOrder.getOrdersStatus(16);
            orders.setOrderStatus(orderStatus);
            serviceCRUDOrder.updateOrder(orders);
//update total products
            System.out.println("data "+
orderProductByOrderId.size());
            for (OrdersProduct ordersProduct :
orderProductByOrderId) {
                Product product = ordersProduct.getProduct()
                System.out.println("data product"+product);
                Integer total = ordersProduct.getQuantity();
                System.out.println("data total"+total);
                Integer quantity = product.getQuantity();
                int i = quantity-total;
                product.setQuantity(i);
                serviceCRUDOrder.saveProduct(product);
                String[] mail = {customer.getEmail()};
                SendMail.sendEMail(mail, Constants.emailSubjectTxt,
Constants.emailMsgTxt, Constants.emailFromAddress); }
                setResponsePage(OrderViewPage.class); } } };
                action.add(statusaction);
                item.add(action);
            } }; return dataView; } }

```

#### ORDERVIEWPAGE.HTML

```

<wicket:extend>
    <div id="tabledata" class="section">
    <h2 class="ico_mug"> DAFTAR PEMESANAN </h2>
    <table id="table">

```

```

<thead><tr>
    <th>ID PEMESANAN</th>
    <th>PEMBELI</th>
    <th>TOTAL PRODUK</th>
    <th>STATUS PEMESANAN</th>
    <th>TINDAKAN</th>
</tr></thead>
<tbody>
<tr wicket:id="users">
<td class="table_date" wicket:id="orderid"></td>
<td class="table_title" wicket:id="customer"></td>
<td class="table_title" wicket:id="total"></td>
<td class="table_title" wicket:id="status"></td>
<td><a href="#" wicket:id="action" title=""
class="vtip"><span wicket:id="statusaction" /></a>
<a href="#" wicket:id="lnkDetail" title=""
class="vtip">DETAIL</a> </td></tr>
</tbody>
</table>
<div class="pagination" style="float: left; width: 800px;">
<span wicket:id="paginator" /></div></div>
</wicket:extend>

```

#### ORDERDETAILVIEWPAGE.JAVA

```

package com.roes.commerce.pages.admin;
import java.util.List;
import org.apache.wicket.PageParameters;
import org.apache.wicket.markup.html.basic.Label;
import org.apache.wicket.markup.html.list.ListItem;
import org.apache.wicket.markup.html.list.PageableListView;
import org.apache.wicket.markup.html.navigation.paging.PagingNavigator;
import org.apache.wicket.model.CompoundPropertyModel;
import org.apache.wicket.model.Model;
import com.roes.commerce.model.Customer;
import com.roes.commerce.model.Orders;
import com.roes.commerce.model.OrdersProduct;
import com.roes.commerce.pages.HomePage;
import com.roes.commerce.service.ServiceCRUDOrder;
public class OrderDetailViewPage extends HomePage {
    private ServiceCRUDOrder serviceCRUDOrder =
ServiceCRUDOrder.getInstance();
    public OrderDetailViewPage(PageParameters
parametersPage) {
        super(parametersPage);
        Integer orderId =
Integer.parseInt(parametersPage.getString("orderId"));
        PageableListView dataView =
createTableData("users", serviceCRUDOrder
.getOrderProductByOrderId(orderId), 10);
        PagingNavigator paginator = new
PagingNavigator("paginator", dataView);
        add(dataView);
        add(paginator); }
    private PageableListView createTableData(String id,
List<OrdersProduct> list, Integer
paging) {
        PageableListView dataView = new
PageableListView(id, list, paging) {
@Override
        protected void populateItem(ListItem item) {

```

```

        final OrdersProduct ordersProduct =
(OrdersProduct) item
        .getModelObject();
        item.setModel(new
CompoundPropertyModel(ordersProduct));
        item.add(new Label("productName",
        new Model(ordersProduct.getProduct().getName())););
        item.add(new Label("harga", new
Model(ordersProduct.getPrice())););

        item.add(new Label("jumlah", new
Model(ordersProduct.getQuantity())););
        item.add(new Label("total", new
Model(ordersProduct.getTotal())); } };
        return dataView; } }

```

### ORDERDETAILVIEWPAGE.HTML

```

<wicket:extend>
    <div id="tabledata" class="section">
        <h2 class="ico_mug">DETAIL PEMESANAN :</h2>
        <!-- <p>
<h5>Order Id : <label wicket:id="orderId"/></h5>
<h5>Nama Customer : <label wicket:id="customerName"/></h5>
<h5>Email : <label wicket:id="email"/></h5>
<h5>Tanggal Transaksi : <label wicket:id="tglTransaksi"/></h5>
</p -->
        <table id="table">
            <thead>
                <tr><th>NAMA PRODUK</th>
                <th>HARGA</th>
                <th>JUMLAH </th>
                <th>TOTAL</th></tr>
            </thead>
            <tbody>
                <tr wicket:id="users">
                    <td class="table_date">
wicket:id="productName"></td>
                    <td class="table_title" wicket:id="harga"></td>
                    <td class="table_title" wicket:id="jumlah"></td>
                    <td class="table_title" wicket:id="total"></td></tr>
                </tbody></table>
<div class="pagination" style="float: left; width: 800px;">
<span wicket:id="paginator" /></div></div>
</wicket:extend>

```

### PRODUCTPAGE.JAVA

```

package com.roes.commerce.pages.admin;
import java.util.Date;
public class ProductPage extends HomePage {
    private ServiceCRUDProduct serviceCRUDProduct =
ServiceCRUDProduct.getInstance();
    private TextField name;
    private TextField qty;
    private TextField price;
    private TextField weight;
    private TextArea desc;
    private DropDownChoice manufacturer;
    private DropDownChoice weightclass;
    private DropDownChoice category;
    private DropDownChoice currency;
    private DateField dateavailable;
    private FileUploadField gbr1;
    private FileUploadField gbr2;
    private FileUploadField gbr3;

```

```

private Form form;
private FeedbackPanel msg;
public ProductPage(PageParameters parametersPage) {
    super(parametersPage);
    name = createTextField("name", false, true);
    qty = (TextField) new TextField("qty",
Integer.class).setRequired(true);
    qty.add(NumberValidator.POSITIVE);
    qty.setModel(new Model());
    price = (TextField) new TextField("price", new Model())
.setRequired(true);
    price.setModel(new Model());
    weight = (TextField) new TextField("weight", Integer.class)
.setRequired(true);
    weight.add(NumberValidator.POSITIVE);
    weight.setModel(new Model());
    desc = new TextArea("desc", new Model());
    if(parametersPage.getString("page")!=null){
        if(parametersPage.getString("page").equals("edit")){
Integer id = Integer.parseInt(parametersPage.getString("id"));
Product product = serviceCRUDProduct.findById(id);
name.setModelObject(product.getName());
qty.setModelObject(product.getQuantity());
price.setModelObject(product.getPrice());
weight.setModelObject(product.getWeight());
desc.setModelObject(product.getDescs()); } }
        IChoiceRenderer iChoiceManufacture =
new IChoiceRenderer() {
            private static final long
serialVersionUID = 1L;
            @Override
            public Object getDisplayValue(Object obj) {
                Manufacture w = (Manufacture) obj;
                return w.getName(); }
            @Override
            public String getIdValue(Object obj, int indeks) {
                Manufacture w = (Manufacture) obj;
                return w.getManufacturerId().toString(); } };
        IChoiceRenderer iChoiceCategory = new IChoiceRenderer() {
            private static final long serialVersionUID = 1L;
            @Override
            public Object getDisplayValue(Object obj) {
                Category w = (Category) obj;
                return w.getCategoryDescription().getName(); }
            @Override
            public String getIdValue(Object obj, int indeks) {
                Category w = (Category) obj;
                return w.getCategoryId().toString(); } };
        IChoiceRenderer iChoiceWeighClass = new IChoiceRenderer() {
            private static final long serialVersionUID = 1L;
            @Override
            public Object getDisplayValue(Object obj) {
                WeightClass w = (WeightClass) obj;
                return w.getTitle(); }
            @Override
            public String getIdValue(Object obj, int indeks) {
                WeightClass w = (WeightClass) obj;
                return w.getWeightClassId().toString(); } };
        IChoiceRenderer iChoiceCurrency = new IChoiceRenderer() {
            private static final long serialVersionUID = 1L;
            @Override

```

```

public Object getDisplayValue(Object obj) {
    Currency w = (Currency) obj;
    return w.getTitle(); }
@Override
public String getIdValue(Object obj, int indeks) {
    Currency w = (Currency) obj;
    return w.getCurrencyId().toString(); } };
manufacturer = (DropDownChoice) new
DropDownChoice("manufacturer",
new Model(), serviceCRUDProduct.getAllManufacture(),
iChoiceManufacture).setRequired(true);
weightclass = (DropDownChoice) new
DropDownChoice("weightclass",
new Model(), serviceCRUDProduct.getAllWeightClass(),
iChoiceWeighClass).setRequired(true);
category = (DropDownChoice) new DropDownChoice("category",
new Model(),serviceCRUDProduct.getAllCategory(),
iChoiceCategory).setRequired(true);
currency = (DropDownChoice) new DropDownChoice("currency",
new Model(),
serviceCRUDProduct.getAllCurrency(), iChoiceCurrency)
.setRequired(true);
dateavailable = new DateField("dateavailable", new Model(new
Date()));
gbr1 = (FileUploadField) new
FileUploadField("gbr1").setRequired(true);
    form = new Form("form") {
        public void onSubmit() {
try {
    Manufacture manufacture = serviceCRUDProduct
.getManufactureById(Integer.parseInt(manufacturer
.getValue()))
    WeightClass weightClass = serviceCRUDProduct
.getWeighClassById(Integer.parseInt(weightclass
.getValue()));
    Category category = serviceCRUDProduct
.getCategoryById(Integer
.parseInt(ProductPage.this.category
.getValue()));
    Currency currency = serviceCRUDProduct
.getCurrencyById(Integer
.parseInt(ProductPage.this.currency
.getValue()));
FileUpload[] fileUploads = {gbr1.getFileUpload};
    Product product = new Product();
    product.setCurrency(currency);
    product.setDateAdded(new Date());
    product.setDateAvailable(dateavailable.getDate());
    product.setImage(serviceCRUDProduct.getBiggestIdPr
oduct()+Constants.extention);
    product.setManufacture(manufacture);
    String replace = price.getValue().replace(",",".");
    product.setPrice(Double.parseDouble(replace));
    product.setQuantity(Integer.parseInt(qty.getValue()));
    product.setStatus(1);
    product.setUser(((WicketSession)getSession()).getUser
());
    product.setWeight(Double.parseDouble(weight.getValue()
e()));
    product.setWeightClass(weightClass);
    product.setDescs(desc.getValue());

    product.setName(name.getValue());
    product.setCategory(category);
    product.setImageBig1(serviceCRUDProduct.getBiggest
IdProduct()+Constants.extention);
    product.setImageThumb1(serviceCRUDProduct.getBig
gestIdProduct()+Constants.extention);
    serviceCRUDProduct.saveProduct(product,
gbr1.getFileUpload());
        } catch (Exception e) {
e.printStackTrace(); } } };
msg = new FeedbackPanel("msg");
Component[] components = { name, qty, price, weight, desc,
manufacturer, weightclass, dateavailable, gbr1
category, currency };
addToForm(form, components);
add(form);
add(msg); } }
public class ProductPage extends HomePage {
    private ServiceCRUDProduct serviceCRUDProduct =
ServiceCRUDProduct
.getInstance();
    private TextField name;
    private TextField qty;
    private TextField price;
    private TextField weight;
    private TextArea desc;
    private DropDownChoice manufacturer;
    private DropDownChoice weightclass;
    private DropDownChoice category;
    private DropDownChoice currency;
    private DateField dateavailable;
    private FileUploadField gbr1;
    private FileUploadField gbr2;
    private FileUploadField gbr3;
    private Form form;
    private FeedbackPanel msg;
    public ProductPage(PageParameters parametersPage) {
        super(parametersPage);
        name = createTextField("name", false, true);
        qty = (TextField) new TextField("qty",
Integer.class).setRequired(true);
        qty.add(NumberValidator.POSITIVE);
        qty.setModel(new Model());
        price = (TextField) new TextField("price",
new Model())
        .setRequired(true);
        price.setModel(new Model());
        weight = (TextField) new
TextField("weight", Integer.class)
        .setRequired(true);
        weight.add(NumberValidator.POSITIVE);
        weight.setModel(new Model());
        desc = new TextArea("desc", new Model());
        if(parametersPage.getString("page")!=null){
            if(parametersPage.getString("page").equals("edit")){
                Integer id = Integer.parseInt(parametersPage.getString("id"));
                Product product = serviceCRUDProduct.findById(id);
                name.setModelObject(product.getName());
                qty.setModelObject(product.getQuantity());
                price.setModelObject(product.getPrice());

```

```

        weight.setModelObject(product.getWeight());
        desc.setModelObject(product.getDescs()); } }
IChoiceRenderer iChoiceManufacture = new IChoiceRenderer() {
    private static final long serialVersionUID = 1L;
@Override
public Object getDisplayValue(Object obj) {
    Manufacture w = (Manufacture) obj;
    return w.getName(); }
@Override
public String getIdValue(Object obj, int indeks) {
    Manufacture w = (Manufacture) obj;
    return w.getManufacturerId().toString(); } };
IChoiceRenderer iChoiceCategory = new IChoiceRenderer() {
    private static final long serialVersionUID = 1L;
@Override
public Object getDisplayValue(Object obj) {
    Category w = (Category) obj;
    return w.getCategoryDescription().getName(); }
@Override
public String getIdValue(Object obj, int indeks) {
    Category w = (Category) obj;
    return w.getCategoryId().toString(); } };
IChoiceRenderer iChoiceWeighClass = new IChoiceRenderer() {
    private static final long serialVersionUID = 1L;
@Override
public Object getDisplayValue(Object obj) {
    WeightClass w = (WeightClass) obj;
    return w.getTitle(); }
@Override
public String getIdValue(Object obj, int indeks) {
    WeightClass w = (WeightClass) obj;
    return w.getWeightClassId().toString(); } };
IChoiceRenderer iChoiceCurrency = new IChoiceRenderer() {
    private static final long serialVersionUID = 1L;
@Override
public Object getDisplayValue(Object obj) {
    Currency w = (Currency) obj;
    return w.getTitle(); }
@Override
public String getIdValue(Object obj, int indeks) {
    Currency w = (Currency) obj;
    return w.getCurrencyId().toString(); } };
manufacturer = (DropDownChoice) new
DropDownChoice("manufacturer",
new Model(), serviceCRUDProduct.getAllManufacture(),
iChoiceManufacture).setRequired(true);
weightclass = (DropDownChoice) new
DropDownChoice("weightclass",
new Model(), serviceCRUDProduct.getAllWeightClass(),
iChoiceWeighClass).setRequired(true);
category = (DropDownChoice) new DropDownChoice("category",
new Model(),
serviceCRUDProduct.getAllCategory(), iChoiceCategory)
.setRequired(true);
currency = (DropDownChoice) new DropDownChoice("currency",
new Model(),
serviceCRUDProduct.getAllCurrency(), iChoiceCurrency)
.setRequired(true);
dateavailable = new DateField("dateavailable", new Model(new
Date()));

```

```

gbr1 = (FileUploadField) new
FileUploadField("gbr1").setRequired(true);
    form = new Form("form") {
        public void onSubmit() {
            try {
                Manufacture manufacture = serviceCRUDProduct
                .getManufactureById(Integer.parseInt(manufacturer
                .getValue()));
                WeightClass weightClass = serviceCRUDProduct
                .getWeighClassById(Integer.parseInt(weightclass
                .getValue()));
                Category category = serviceCRUDProduct
                .getCategoryById(Integer
                .parseInt(ProductPage.this.category
                .getValue()));
                Currency currency = serviceCRUDProduct
                .getCurrencyById(Integer
                .parseInt(ProductPage.this.currency
                .getValue()));
                FileUpload[] fileUploads = {gbr1.getFileUpload};
                Product product = new Product();
                product.setCurrency(currency);
                product.setDateAdded(new Date());
                product.setDateAvailable(dateavailable.getDate());
                product.setImage(serviceCRUDProduct.getBiggestIdPr
                oduct()+Constants.extention);
                product.setManufacture(manufacture);
                String replace = price.getValue().replace(",",".");
                product.setPrice(Double.parseDouble(replace));
                product.setQuantity(Integer.parseInt(qty.getValue()));
                product.setStatus(1);
                product.setUser(((WicketSession)getSession()).getUser
                ());
                product.setWeight(Double.parseDouble(weight.getValu
                e()));
                product.setWeightClass(weightClass);
                product.setDescs(desc.getValue());
                product.setName(name.getValue());
                product.setCategory(category);
                product.setImageBigI(serviceCRUDProduct.getBiggest
                IdProduct()+Constants.extention);
                product.setImageThumbI(serviceCRUDProduct.getBig
                gestIdProduct()+Constants.extention);
                serviceCRUDProduct.saveProduct(product,
                gbr1.getFileUpload());
            } catch (Exception e) {
                e.printStackTrace(); } } };
msg = new FeedbackPanel("msg");
Component[] components = { name, qty, price, weight, desc,
    manufacturer, weightclass, dateavailable, gbr1,
        category, currency };
    addToForm(form, components);
    add(form);
    add(msg); } }

```

#### PRODUCTPAGE.HTML

```

<wicket:extend>
<div id="tabledata" class="section">
<span wicket:id="msg" style="float: left;clear: both;"></span>
<form wicket:id="form">
<table border="0">

```

```

<tr><td>NAMA:</td>
<td><input type="text" wicket:id="name"/></td></tr>
<tr><td>JUMLAH:</td>
<td><input type="text" wicket:id="qty"/></td></tr>
<tr><td>HARGA:</td>
<td><input type="text" wicket:id="price"/></td></tr>
<tr><td>BERAT:</td>
<td><input type="text" wicket:id="weight"/></td></tr>
<tr><td>KATEGORI:</td>
<td><select wicket:id="category"></select></td></tr>
<tr><td>KETERANGAN:</td>
<td><textarea wicket:id="desc"></textarea></td></tr>
<tr><td>MANUFAKTUR:</td>
<td><select wicket:id="manufacturer"></select></td></tr>
<tr><td>SATUAN BERAT:</td>
<td><select wicket:id="weightclass"></select></td></tr>
<tr><td>TANGGAL DITAMBAHKAN:</td>
<td><span wicket:id="dateavailable"/></td></tr>
<tr><td>Gambar 1:</td><td>
<input type="file" wicket:id="gbr1"/></td></tr>
<tr><td>MATA UANG:</td>
<td><select wicket:id="currency"></select></td></tr>
<tr><td colspan="2" align="center">
<input type="submit" value="SIMPAN">
<td colspan="2" align="center">
<input type="reset" value="RESET"></td></tr>
</table></form></div>
</wicket:extend>

```

#### PRODUCTVIEWPAGE.JAVA

```

package com.roes.commerce.pages.admin;
import java.io.File;
public class ProductViewPage extends HomePage {
    private ServiceCRUDProduct serviceCRUDProduct =
ServiceCRUDProduct.getInstance();
    private Label name;
    private Label weight;
    private Label price;
    private Label category;
    private Label manufactur;
    private String path =
System.getProperty("catalina.base")
+ File.separator + "webapps" + File.separator
+ Constants.projectName + File.separator + "images";
    public ProductViewPage(PageParameters
parametersPage) throws Exception {
        super(parametersPage);
        List<Product> all =
serviceCRUDProduct.getAll();
        PageableListView dataView =
createTableData("products", all, 10);
        BookmarkablePageLink linknew = new
BookmarkablePageLink("linknew",
ProductPage.class, new PageParameters());
        PagingNavigator paginator = new
PagingNavigator("paginator", dataView);
        add(dataView);
        add(linknew);
        add(paginator); }
    private PageableListView createTableData(String
string, List<Product> all,int i) throws Exception{

```

```

        PageableListView listView = new
PageableListView(string, all, i) {
    @Override
    protected void populateItem(ListItem item) {
        try {
            final Product product = (Product) item.getModelObject();
            String name2 = product.getName();
            name = new Label("name", new Model(name2));
            String title = product.getWeightClass().getTitle();
            String string2 = product.getWeight().toString();
            weight = new Label("weight", new Model(string2 + " "
+ title));
            String title2 = product.getCurrency().getTitle();
            price = new Label("price", new
Model(product.getPrice()+ " " + title2));
            String name3 = product.getCategory()
.getCategoryDescription().getName();
            category = new Label("category", new Model(name3));
            manufactur = new Label("manufactur", new
Model(product.getManufacture().getName()));
            AjaxFallbackLink lnkdelete = new AjaxFallbackLink(
"lnkdelete") {
                @Override
                public void onClick(AjaxRequestTarget target) {
                    try {
                        serviceCRUDProduct.deleteProduct(product);
                        setResponsePage(ProductViewPage.class);
                    } catch (Exception e) {
                        e.printStackTrace(); } } };
            AjaxFallbackLink lnkedit = new
AjaxFallbackLink("lnkedit") {
                @Override
                public void onClick(AjaxRequestTarget target) {
                    PageParameters parameters = new PageParameters();
                    parameters.add("page", "edit");
                    parameters.add("id", product.getProductId().toString());
                    setResponsePage(ProductPage.class, parameters); } } ;
            Label available = new Label("available", new
Model(product.getQuantity()));
            Label label[] = { name, weight, price, category,
manufactur,available };
            addLabelToItem(label, item);
            item.add(lnkdelete);
            item.add(lnkedit);
        } catch (Exception e) {
            e.printStackTrace(); } } };
        return listView; } }

```

#### PRODUCTVIEWPAGE.HTML

```

<wicket:extend>
<div id="tabledata" class="section">
<h2 class="ico_mug">DAFTAR PRODUK</h2>
<a wicket:id="linknew" style="float: left; clear: both;">BUAT
BARU</a>
<table id="table">
<thead>
<tr>
<th>NAMA</th>
<th>HARGA</th>
<th>BERAT</th>

```

```

<th>JUMLAH</th>
<th>KATEGORI</th>
<th>MANUFAKTUR</th>
<th>TINDAKAN</th></tr></thead>
<tbody>
<tr wicket:id="products">
<td class="table_title" wicket:id="name"></td>
<td class="table_title" wicket:id="price"></td>
<td class="table_title" wicket:id="weight"></td>
<td class="table_title" wicket:id="available"></td>
<td class="table_title" wicket:id="category"></td>
<td class="table_title" wicket:id="manufacturer"></td>
<td><a href="#" wicket:id="Inkdelete" title="delete this
PRODUCT" class="vtip">delete</a>
<a href="#" wicket:id="Inkedit" title="edit this book"
class="vtip">edit</a></td></tr></tbody></table>
<div class="pagination" style="float: left; width: 800px;">
<span wicket:id="paginator" /></div></div>
</wicket:extend>

```

### USERPAGE.JAVA

```

package com.roes.commerce.pages.admin;
import java.util.Date;
public class UserPage extends HomePage{
    private ServiceCRUDUser serviceCRUDUser =
ServiceCRUDUser.getInstance();
    public UserPage(PageParameters parameters){
        super(parameters);
        final String string = parameters.getString("page");
        final String string2 = parameters.getString("id");
        Form form = new Form("form");
final TextField firstname = createTextField("firstname", false,
false);
final TextField lastname = createTextField("lastname", false,
false);
final TextField username = createTextField("username", false,
false);
final TextField email = createTextField("email", false, false);
        final PasswordTextField password =new
        PasswordTextField("password",new Model());
        final PasswordTextField repassword =new
        PasswordTextField("repassword",new Model());
final DropDownChoice usergroup =
createDropDownChoice("usergroup", false, false,
serviceCRUDUser.getAllUserGroup());
        if(string!=null && string2 !=null){
            if (string.equals("edit")) {
                User user = serviceCRUDUser.findById(Integer.parseInt(string2));
                firstname.setModelObject(user.getFirstname());
                lastname.setModelObject(user.getLastname());
                username.setModelObject(user.getUsername());
                email.setModelObject(user.getEmail());
                usergroup.setModelObject(user.getUserGroup().getNa
                me());
                password.setModelObject(user.getPassword()); } }
        AjaxButton btnSave = new AjaxButton("btnSave"){
@Override
protected void onSubmit(AjaxRequestTarget target, Form form) {
        if(firstname.getValue().isEmpty()||lastname.getValue().
isEmpty()||username.getValue().isEmpty()||email.getValue().isEmp
ty()||usergroup.getValue().equals("-1")){

```

```

target.appendJavascript("alert('data harus complete')");
        }else{
            if(password.getValue().equals(repassword.getValue()))
{String namaUserGroup =
serviceCRUDUser.getAllUserGroup().get(Integer.parseInt(usergro
up.getValue()));
            UserGroup userGroup =
serviceCRUDUser.getUserGroupByName(namaUserGroup);
            User user = null;
            if(string!=null){
                if(string.equals("edit"))
user = serviceCRUDUser.findById(Integer.parseInt(string2));
            }else{
                user = new User(); }
            user.setDateAdded(new Date());
            user.setEmail(email.getValue());
            user.setFirstname(firstname.getValue());
            user.setLastname(lastname.getValue());
            user.setPassword(new
MD5(password.getValue()).toString());
            user.setStatus(1);
            user.setUsername(username.getValue());
            serviceCRUDUser.saveUser(user, userGroup);
            target.appendJavascript("alert('sukses simpan')");
            //setResponsePage(UserPage.class);
        }else{
            target.appendJavascript("alert('password harus sama')");
        } } } };

        Component[] components = {firstname, lastname,
username, email, password, repassword, btnSave, usergroup};
        addToForm(form, components);
        add(form); } }

```

### USERPAGE.HTML

```

<wicket:extend>
<div id="tabledata" class="section">
<form wicket:id="form" >
<table border="0">
<tr><td>NAMA ADMIN:</td>
<td><input type="text" wicket:id="username"></td></tr>
<tr><td>NAMA:</td>
<td><input type="text" wicket:id="firstname"></td></tr>
<tr><td>NAMA LENGKAP:</td>
<td><input type="text" wicket:id="lastname"></td></tr>
<tr><td>EMAIL:</td>
<td><input type="text" wicket:id="email"></td></tr>
<tr><td>PASSWORD:</td>
<td><input type="password" wicket:id="password"></td></tr>
<tr><td>RE-PASSWORD:</td>
<td><input type="password" wicket:id="repassword"></td></tr>
<tr><td>GRUP:</td>
<td><select wicket:id="usergroup"/></td></tr>
<tr><td colspan="2" align="center"><input type="submit"
value="SIMPAN" wicket:id="btnSave"></td>
<td colspan="2" align="center"><input type="reset"
value="RESET"></td></tr></table></form></div>
</wicket:extend>

```

### USERVIEWPAGE.JAVA

```

package com.roes.commerce.pages.admin;

```

```

import java.util.List;
public class UserViewPage extends HomePage {
    private ServiceCRUDUser serviceCRUDUser =
ServiceCRUDUser.getInstance();
    public UserViewPage(PageParameters parametersPage)
{
    super(parametersPage);
    PageableView dataView = createTableData("users",
serviceCRUDUser.getAll(), 10);
    PagingNavigator paginator = new
PagingNavigator("paginator", dataView);
    BookmarkablePageLink linknew = new
BookmarkablePageLink("linknew",
UserPage.class.new PageParameters());
    add(linknew);
    add(dataView);
    add(paginator); }
    private PageableView createTableData(String id,
List list, Integer paging) {
PageableView dataView = new PageableView(id, list,
paging) {
    @Override
protected void populateItem(ListItem item) {
    final User user = (User) item.getModelObject();
    item.setModel(new CompoundPropertyModel(user));
    item.add(new Label("username", new
Model(user.getUsername())));
    item.add(new Label("firstname", new Model(user
.getFirstname())));
    item.add(new Label("lastname", new
Model(user.getLastname())));
    item.add(new Label("email", new
Model(user.getEmail())));
    AjaxFallbackLink lnkdelete = new
AjaxFallbackLink("lnkdelete") {
    @Override
public void onClick(AjaxRequestTarget target) {
    try {
        serviceCRUDUser.deleteUser(user);
        setResponsePage(UserViewPage.class);
    } catch (Exception e) {
e.printStackTrace(); } } };
    AjaxFallbackLink lnkedit = new AjaxFallbackLink("lnkedit") {
    @Override
public void onClick(AjaxRequestTarget target) {
        PageParameters parameters = new PageParameters();
        parameters.add("page", "edit");
        parameters.add("id", user.getId().toString());
        setResponsePage(UserPage.class, parameters); } };
    item.add(lnkdelete);
    item.add(lnkedit); } };
return dataView; } }

```

#### USERVIEWPAGE.HTML

```

<wicket:extend>
<div id="tabledata" class="section">
<h2 class="ico_mug">DAFTAR USER</h2>
<a wicket:id="linknew" style="float: left; clear: both;">BUAT
BARU</a>
<table id="table"
<thead>

```

```

<tr>
<th>NAMA ADMIN</th>
<th>NAMA</th>
<th>NAMA LENGKAP</th>
<th>EMAIL</th>
<th>TINDAKAN</th></tr></thead>
<tbody>
<tr wicket:id="users">
<td class="table_date" wicket:id="username"></td>
<td class="table_title" wicket:id="firstname"></td>
<td class="table_title" wicket:id="lastname"></td>
<td class="table_title" wicket:id="email"></td>
<td><a href="#" wicket:id="lnkdelete" title="delete this book"
class="vtip">delete</a>
<a href="#" wicket:id="lnkedit" title="edit this book"
class="vtip">edit</a></td></tr></tbody></table>
<div class="pagination" style="float: left; width: 800px;">
<span wicket:id="paginator" /></div></div>
</wicket:extend>

```

#### CUSTOMER

##### CSS CUSTOMER STYLE.CSS

```

body
{
background-color:#6F605A;
font-family:Arial, Helvetica, sans-serif;
padding:0;
font-size:12px;
margin:0px auto auto auto;
color:#000000;
}
a{
color:#f8981d;
}
.clear{
clear:both;
}
p{
padding:5px 0 5px 0;
margin:0px;
text-align:justify;
line-height:19px;
}
p.details{
padding:5px 15px 5px 15px;
font-size:11px;
}
p.details_cart{
clear:both;
padding:25px 30px 5px 0px;
font-size:11px;
font-style:italic;
}
p.more_details{
padding:25px 20px 0px 20px;
font-size:11px;
}
#wrap{

```

```

width:900px;
height: auto;
margin:auto;
background-color:#FFFFFF;
}
.header{
width:900px;
height:181px;
background:url(images/header.jpg) no-repeat center;
}
.logo{
padding:50px 0 0 130px;
}
/*-----menu-----*/
#menu{
width:628px;
height:30px;
padding:55px 0 0 10px;
}
#menu ul{
display:block;
list-style:none;
padding:9px 0 0 10px;
margin:0px;
}
#menu ul li{
display:inline;
padding:0px;
margin:0px;
height:20px;
}
#menu ul li a{
height:27px;
display:block;
padding:0px 10px 0 10px;
margin:0 4px 0 4px;
margin-left:2px;
float:left;
text-decoration:none;
text-align:center;
color:#fff;
font-size:13px;
line-height:25px;
}
#menu ul li.selected a{
height:27px;
display:block;
padding:0px 10px 0 10px;
margin:0 5px 0 5px;
float:left;
text-decoration:none;
text-align:center;
color:#f7a849;
font-size:13px;
line-height:25px;
}
#menu ul li a:hover{
color:#f7a849;
}
/*-----crumb_nav-----*/

.crumb_nav{
padding:5px 0 10px 0px;
}
.crumb_nav a{
color:#990000;
}
span.red{
color:#990000;
}
.price{
font-size:14px;
padding:0 0 0 15px;
margin:2px 0 5px 0;
}
span.colors{
padding:2px 2px 0 2px;
}
/*-----center content-----*/
.center_content{
width:900px;
padding:0px 0 0 0;
background:url(images/center_bg.gif) repeat-y center;
}
.left_content{
width:490px;
float:left;
padding:20px 0 20px 20px;
}
.right_content{
width:370px;
float:left;
padding:20px 0 20px 20px;
}
.title{
color:#734633;
padding:0px;
float:left;
font-size:19px;
margin:10px 0 10px 0;
}
span.title_icon{
float:left;
padding:0 5px 0 0;
}
.prod_title{
color:#f8981d;
padding:5px 0 0 15px;
font-size:13px;
}
a.more{
font-style:italic;
color:#f8981d;
float:right;
text-decoration:none;
font-size:11px;
padding:0px 15px 0 0 ;
}
.about{
width:337px;
clear:both;
background:url(images/border.gif) no-repeat bottom center;
}

```



```

padding:0 0 35px 0;
}
img.right{
float:right;
padding:0 0 0 10px;
}
.right_box{
width:170px;
float:left;
padding:10px 0 0 0;
}
/*-----feat_prod_box-----*/
.feat_prod_box{
padding:10px 0 10px 10px;
margin:0 20px 20px 0;
border-bottom:1px #b2b2b2 dashed;
clear:both;
}
.feat_prod_box_details{
padding:10px 0 10px 0;
margin:0 20px 10px 0;
clear:both;
}
.prod_img{
float:left;
padding:0 5px 0 0;
text-align:center;
}
.prod_det_box{
width:295px;
float:left;
padding:0 0 0 25px;
position:relative;
}
.box_top{
width:295px;
height:9px;
background:url(images/box_top.gif) no-repeat center bottom;
}
.box_center{
width:295px;
height:auto;
background:url(images/box_center.gif) repeat-y center;
}
.box_bottom{
width:295px;
height:9px;
background:url(images/box_bottom.gif) no-repeat center top;
}
.new_prod_box{
float:left;
text-align:center;
padding:10px;
}
.new_prod_box a{
padding:5px 0 5px 0;
color:#b5b5b6;
text-decoration:none;
display:block;
}
.new_prod_bg{

```

```

width:132px;
height:119px;
text-align:center;
background:url(images/new_prod_box.gif) no-repeat center;
position:relative;
}
.new_icon{
position:absolute;
top:0px;
right:0px;
z-index:200;
}
.special_icon{
position:absolute;
top:0px;
_top:6px;
right:2px;
z-index:250;
}
img.thumb{
padding:10px 0 0 0;
}
.new_products{
clear:both;
padding:0px;
}

ul.list{
clear:both;
padding:10px 0 0 20px;
margin:0px;
}
ul.list li{
list-style:none;
padding:2px 0 2px 0;
}
ul.list li a{
list-style:none;
text-decoration:none;
color:#000000;
background:url(images/left_menu_bullet.gif) no-repeat left;
padding:0 0 0 17px;
}
ul.list li a:hover{
text-decoration:underline;
}
/* demo */
div.demolayout {
width:460px;
margin: 0 0 20px 0;
}
ul.demolayout {
list-style-type: none;
float: left;
margin:0px;
padding:0px;
}

ul.demolayout li {
margin: 0 10px 0 0;
float: left;

```

```

}
.tab{
border:1px #DFDFDF solid;
padding:0 0 25px 0;
}
ul.demolayout a {
float: left;
display: block;
padding: 5px 25px;
border: 1px solid #DFDFDF;
border-bottom: 0;
color: #666;
background: #eee;
text-decoration: none;
font-weight: bold; }
ul.demolayout a:hover {
background: #fff; }
ul.demolayout a.active {
background: #fff;
padding-bottom: 5px;
cursor: default;
color:#795636;
}
.tabs-container {
clear: left;
padding:0px;
}
/*-----languages_box-----*/
.languages_box {
padding:0 0 5px 0;
float:left;
}
.languages_box a{
padding:0 2px 0 2px;
}
.languages_box a.selected{
padding:2px 2px 0 2px;
border:1px #CCCCCC solid;
}
.currency{
float:left;
padding:0 0 0 20px;
}
.currency a{
text-decoration:none;
color:#333333;
padding:3px;
border:1px #eeedee solid;
}
.currency a.selected{
text-decoration:none;
color:#fff;
padding:3px;
border:1px #eeedee solid;
background-color:#FF9900;
font-weight:bold;
}
.currency a:hover{
border:1px #990000 solid;
}
/*-----cart-----*/

```

```

.cart{
width:337px;
float:left;
height:40px;
margin:10px 0 10px 0;
background:url(images/border.gif) no-repeat bottom center;
padding:0 0 30px 0;
}
.home_cart_content{
float:left;
padding:3px;
border:1px #eeedee solid;
margin:10px 0 0 15px;
}
a.view_cart{
display:block;
float:left;
margin:12px 0 0 10px;
color:#990000;
}

/*-----contact_form-----*/
.contact_form{
width:355px;
float:left;
padding:25px;
margin:20px 0 0 15px;
margin-left:20px;
border:1px #DFD1D2 dashed;
position:relative;
}
.form_row{
width:335px;
width:355px;
clear:both;
padding:10px 0 10px 0;
padding-left:5px;
color:#a53d17;
}
label.contact{
width:75px;
float:left;
font-size:12px;
text-align:right;
padding:4px 5px 0 0;
color: #333333;
}
input.contact_input{
width:253px;
height:18px;
background-color:#fff;
color:#999999;
border:1px #DFDFDF solid;
float:left;
}
textarea.contact_textarea{
width:253px;
height:120px;
font-family:Arial, Helvetica, sans-serif;
font-size:12px;
color: #999999;
}

```

```

background-color:#fff;
border:1px #DFDFDF solid;
float:left;
}
input.register{
width:71px;
height:25px;
border:none;
cursor:pointer;
text-align:center;
float:right;
color:#FFFFFF;
background:url(images/register_bt.gif) no-repeat center;
}
a.contact{
width:53px;
height:24px;
display:block;
float:right;
margin:0 0 0 10px;
background:url(images/contact_bt.gif) no-repeat center;
text-decoration:none;
text-align:center;
line-height:24px;
color:#fff;
}
a.checkout{
width:71px;
height:25px;
display:block;
float:right;
margin:10px 30px 0 10px;
background:url(images/register_bt.gif) no-repeat center;
text-decoration:none;
text-align:center;
line-height:25px;
color:#fff;
}
a.continue{
width:71px;
height:25px;
display:block;
float:left;
margin:10px 0 0 0px;
background:url(images/register_bt.gif) no-repeat center;
text-decoration:none;
text-align:center;
line-height:25px;
color:#fff;
}
.terms{
padding:0 0 0 80px;
}
.form_subtitle{
position:absolute;
top:-11px;
left:7px;
width:auto;
height:20px;
background-color:#795636;
text-align:center;

```

```

padding:0 7px 0 7px;
color:#FFFFFF;
font-size:11px;
line-height:20px;
}
/*-----cart_table-----*/
.cart_table{
width:440px;
border:1px #CCCCCC solid;
text-align:center;
}
tr.cart_title{
background-color:#DFDFDF;
}
td{
padding:3px;
}
td.cart_total{
text-align:right;
padding:5px 15px 5px 0;
}
img.cart_thumb{
border:1px #b2b2b2 solid;
padding:2px;
}
/*-----*/
div.pagination {
width:420px;
padding:5px;
margin:5px;
text-align:center;
float:left;
clear:both;
font-size:10px;
}
div.pagination a {
padding: 2px 5px 2px 5px;
margin-right: 2px;
border: 1px solid #795636;
text-decoration: none;
color: #795636;
}
div.pagination a:hover, div.pagination a:active {
border: 1px solid #795636;
color: #fff;
background-color: #795636;
}
div.pagination span.current {
padding: 2px 5px 2px 5px;
margin-right: 2px;
border: 1px solid #795636;
font-weight: bold;
background-color: #795636;
color: #FFF;
}
div.pagination span.disabled {
padding: 2px 5px 2px 5px;
margin-right: 2px;
border: 1px solid #f3f3f3;
color: #ccc;
}

```

```

/*-----footer-----*/
.footer{
height:100px;
border-top:1px #b2b2b2 dashed;
background:url(images/footer_bg.gif) no-repeat bottom;
}
.left_footer{
float:left;
padding:10px 0 0 10px;
}
.right_footer{
float:right;
padding:10px 10px 0 0;
}
.footer a{
text-decoration:none;
padding:0 5px 0 5px;
color:#afaef;
}

/*-----*/
#table{width:100%;border:1px solid #c3c3c3;}
#table th{background:#eee;padding:5px;text-align:left;}
#table td{padding:5px;}
#table .table_date{color:#5e5e5e;width:100px;font-weight:bold;}
#table .table_check{width:20px;background:#eee;}
#table .table_title a{color:#26608e;text-decoration:underline;font-weight:bold;}
#table_options{background:#fff;border:solid #c3c3c3;border-width:0px 1px 1px 1px;padding:5px;}
#table_options label{margin:0px;padding-top:2px;}
#table_options ul{list-style:none;margin:0px;padding:0px;}
#table_options ul li{float:left;padding:5px;margin-right:10px;}

table {border-collapse: collapse;border-spacing: 0;}

```

## LIGHTBOX.CSS

```

#lightbox{
position: absolute;
left: 0;
width: 100%;
z-index: 100;
text-align: center;
line-height: 0;
}
#lightbox a img{ border: none; }
#outerImageContainer{
position: relative;
background-color: #fff;
width: 250px;
height: 250px;
margin: 0 auto;
}
#imageContainer{
padding: 10px;
}
#loading{
position: absolute;
top: 40%;
left: 0%;
height: 25%;

```

```

width: 100%;
text-align: center;
line-height: 0;
}
#hoverNav{
position: absolute;
top: 0;
left: 0;
height: 100%;
width: 100%;
z-index: 10;
}
#imageContainer>#hoverNav{ left: 0;}
#hoverNav a{ outline: none;}
#prevLink, #nextLink{
width: 49%;
height: 100%;
background: transparent url(../images/blank.gif) no-repeat; /* Trick IE into showing hover */
display: block;
}
#prevLink { left: 0; float: left;}
#nextLink { right: 0; float: right;}
#prevLink: hover, #prevLink: visited: hover { background: url(../images/prevlabel.gif) left 15% no-repeat; }
#nextLink: hover, #nextLink: visited: hover { background: url(../images/nextlabel.gif) right 15% no-repeat; }
#imageDataContainer{
font: 10px Verdana, Helvetica, sans-serif;
background-color: #fff;
margin: 0 auto;
line-height: 1.4em;
overflow: auto;
width: 100%
}
#imageData{ padding:0 10px; color: #666; }
#imageData #imageDetails{ width: 70%; float: left; text-align: left; }
#imageData #caption{ font-weight: bold; }
#imageData #numberDisplay{ display: block; clear: left; padding-bottom: 1.0em; }
#imageData #bottomNavClose{ width: 66px; float: right; padding-bottom: 0.7em; }
#overlay{
position: absolute;
top: 0;
left: 0;
z-index: 90;
width: 100%;
height: 500px;
background-color: #000;
}

```

## ABOUT.JAVA

```

package com.roes.commerce.pages.frontpage;
import java.util.List;
public About() throws Exception{
PageableListView categories =
createListCategories("categories",
serviceCRUDProduct.getAllCategory(), 10);
add(categories); }

```

## ABOUT.HTML

```
</wicket:extend>
<div class="center_content">
<div class="left_content">
<div class="title">
<span class="title_icon"> </span>TENTANG KAMI</div>
<div class="feat_prod_box_details">
<p class="details">

TENTANG KAMI · Melayani penjualan secara
online/website Aurora.com sejak 14 April 2010 Aurora.com
menjual peralatan komputer, terutama lebih terkait ke Personal
Komputer, Projector, Baterai, Adaptor, Accesories dan lainnya.
Dengan menggunakan TIKI JNE sebagai pelayanan pengiriman,
kami : · Menjangkau penjualan ke seluruh wilayah Indonesia. ·
Terpercaya karena berkerjasama dengan TIKI JNE sehingga kami
mendapatkan prioritas pengiriman. Dengan menggunakan internet
banking, kami melayani pembayaran melalui Bank BNI. Kami bisa
mengecek secara realtime bahkan di hari Sabtu, sehingga
pengiriman Anda tidak terlambat dan tetap bisa kami proses pada
akhir pekan. Alasan untuk memilih kami :
· Harga PAS dan termurah.
· Hand to hand, memotong banyak rantai penjualan yang selama ini
terjadi di toko konvensional.
· Harga termurah bukan berarti mengorbankan kualitas
· Pelayanan e-mail yang cepat respon.
· Melayani pengiriman seluruh Indonesia. · Barang lengkap dan
banyak.
· 99% dari produk yang tampil di website adalah READY STOCK.
```

Website kami www.Aurora.com tersedia 24 jam sehari 7 hari seminggu non stop untuk melayani Anda dalam keperluan mendadak kapanpun, dimanapun. Jam operasi workshop : Senin-Sabtu 09.00 – 16.30 WIB

Minggu/Libur Tutup Alamat kami : Menco Raya 16 Pabelan Kartasura Surakarta, 57102 Telp : 0271-727899, 0271-2019657 Khusus SMS : 0271-9367510 YM: aurora.solo Perhatian: Aurora.com TIDAK ada cabang, kami tidak pernah berafiliasi ataupun mengambil barang dari supplier lokal, karena semua barang kami import langsung secara sendiri dan menjual langsung ke customer secara langsung / DIRECT SELLING</p></div>

```
<div class="clear"></div></div>
<div class="right_content">
<div class="cart">
<div class="title"><span class="title_icon"> </span>My cart</div>
<div class="home_cart_content">3 x items |
<span class="red">TOTAL: 100$</span></div>
<a href="cart.html" class="view_cart">view cart</a></div>

<div class="title"><span class="title_icon"> </span>About Our Store
</div>
<div class="about"><p>

Aurora.com menjual peralatan komputer, terutama
lebih terkait ke Personal Komputer, Projector, Baterai, Adaptor,
Accesories dan lainnya.</p></div>
<div class="clear"></div></div>
```

```
</wicket:extend>
```

## CART.JAVA

```
package com.roes.commerce.pages.frontpage;
import java.io.File;
public class Cart extends FrontPage {
public WebMarkupContainer cart = (WebMarkupContainer) new
WebMarkupContainer("cart").setOutputMarkupId(true);
private Map<Product, Integer> mapProducts = new
HashMap<Product, Integer>();
private double totalPrice = 0.0;
private Set<Product> setProduct = new HashSet<Product>();
public Cart() throws Exception { PageableListView categories =
createListCategories("categories",
serviceCRUDProduct.getAllCategory(), 10); add(categories);
cart.add(new Label("numberItems", new
Model(getNumberItems())));
cart.add(new Label("totalItems", new Model(getTotalItems())));
cart.add(new AjaxFallbackLink("viewCart") {
@Override public void onClick(AjaxRequestTarget target) {
setResponsePage(Cart.class); } });
add(cart);
List<Product> listProductSession = ((WicketSession) getSession())
.getProducts();
for (Product product : listProductSession) {
int countProductItem = 0; for (Product product2 :
listProductSession) {
if (product2.equals(product)) { countProductItem++; }}
mapProducts.put(product, countProductItem); }
setProduct = new HashSet<Product>(listProductSession);
System.out.println("jumlah data "+setProduct.size());
add(createProductTable("productTable", new ArrayList<Product>(
setProduct), 100));
for (Product product2 : listProductSession) {
totalPrice += product2.getPrice();}
add(new Label("total", new Model(totalPrice)));
add(new PageLink("continue", Index.class));
add(new AjaxFallbackLink("checkout")){
@Override
public void onClick(AjaxRequestTarget target) {
if(setProduct.size()==0){
target.appendJavascript("alert('Barang Belanjaan anda masih
kosong, silahkan isi terlebih dahulu'); }else{
((WicketSession) getSession()).setMapProduct(mapProducts);
((WicketSession) getSession()).setSetProduct(setProduct);
setResponsePage(Checkout.class); } } });
public PageableListView createProductTable(String id, final
List<Product> products, Integer max) {
PageableListView pageableListView = new PageableListView(id,
products, max) {
@Override
protected void populateItem(ListItem item) { final Product
productItem = (Product) item.getModelObject();
System.out.println("size " + mapProducts.size());
if (mapProducts.size() > 0) {
Integer count = (Integer) mapProducts.get(productItem);
item.add(new Label("name", new Model(productItem
.getName()))); item.add(new Label("price", new
Model(productItem.getPrice())));
item.add(new Label("qty", new Model(count)));
item.add(new Label("total",
```

```

new Model(productItem.getPrice()* count));
String string2 = relatifpath + File.separator+
productItem.getProductld() + File.separator+ "thumbnail" +
File.separator + "1.jpg"; Image imgThumb = null;
try { imgThumb = new Image("imgThumb", getImage(string2));
catch (IOException e) {
// TODO Auto-generated catch block
e.printStackTrace(); AjaxFallbackLink lnkDel = new
AjaxFallbackLink("lnkDel"){
@Override
public void onClick(AjaxRequestTarget target) {
((WicketSession)
getSession()).getProducts().remove(productItem);setResponsePage
(Cart.class); } };
item.add(imgThumb); item.add(lnkDel);
//mapProducts.remove(productItem); } } };
return pageableListView; } }

```

### CART.HTML

```

<wicket:extend>
<div class="center_content">
<div class="left_content">
<div class="title">
<span class="title_icon"></span>KERANJANG</div>
<div class="feat_prod_box_details">
<table class="cart_table" >
<tr class="cart_title">
<td>gambar</td>
<td>nama produk</td>
<td>harga</td>
<td>jumlah</td>
<td>total</td>
<td>Hapus</td></tr>
<tr wicket:id="productTable">
<td><a href="#"><img wicket:id="imgThumb"
alt="" title="" border="0" class="cart_thumb" /></a></td>
<td wicket:id="name">Komputer</td>
<td wicket:id="price">100$</td>
<td wicket:id="qty">1</td>
<td wicket:id="total">100$</td>
<td wicket:id="lnkDel"><a style="cursor:
pointer;">Delete</a></td></tr>
<tr><td colspan="4" class="cart_total">
<span class="red">TOTAL:</span></td>
<td><label wicket:id="total">total</label></td></tr></table>
<a href="#" class="continue" wicket:id="continue">&lt;
continue</a>
<a href="#" class="checkout" wicket:id="checkout">checkout
&gt;</a></div>
<div class="clear"></div></div>
<div class="right_content">
<div class="cart" wicket:id="cart">
<div class="title"><span class="title_icon"></span>pesanan</div>
<div class="home_cart_content">
<label wicket:id="numberItems">3 x Items</label>
<span class="red" wicket:id="totalItems">TOTAL:
100$</span></div>

```

```

<a href="#" class="view_cart" wicket:id="viewCart">view
cart</a></div>
<div class="title"><span class="title_icon"></span>aurora
komputer</div>
<div class="about">
<p>
Aurora.com menjual peralatan komputer, terutama
lebih terkait ke Personal Komputer, Projector, Baterai, Adaptor,
Accessories dan lainnya.</p></div>
<div class="title">
<span class="title_icon"></span>KATEGORI</div>
<ul class="list" wicket:id="categories">
<li><a wicket:id="lnkuser" style="cursor: pointer"> <label
wicket:id="lblUsername">[categoryname]</label> </a></li></ul>
</div></div>
<div class="clear"></div>
</wicket:extend>

```

### CATEGORY.JAVA

```

package com.roes.commerce.pages.frontpage;
import java.io.File;
public class Category extends FrontPage {
public WebMarkupContainer cart = new
WebMarkupContainer("cart");
public Category(PageParameters parametersPage) throws
Exception { PageLink lnkHome = new PageLink("lnkHome",
Index.class);
add(lnkHome); if (parametersPage.getString("page") != null) {
if (parametersPage.getString("page").equals("edit")) {
Integer id = Integer.parseInt(parametersPage.getString("id"));
com.roes.commerce.model.Category category =
serviceCRUDProduct.getCategoryById(id);
Label categoryName = new Label("categoryName", new Model(
category.getCategoryDescription().getName()));
add(categoryName);
List<Product> products = serviceCRUDProduct.
getProductsByCategoryId(id);
PageableListView productsView = createTableData("productCat",
products, 10); add(productsView);
PagingNavigator paginator = new PagingNavigator("paginator",
productsView);
add(paginator); } }
PageableListView categories = createListCategories("categories",
serviceCRUDProduct.getAllCategory(), 10);
add(categories);
cart.add(new Label("numberItems", new
Model(getNumberItems())));
cart.add(new Label("totalItems", new Model(getTotalItems())));
cart.add(new AjaxFallbackLink("viewCart") {
@Override
public void onClick(AjaxRequestTarget target) {
setResponsePage(Cart.class);}});
add(cart); }
private PageableListView createTableData(String string,
List<Product> all, int i) throws Exception {
PageableListView listView = new PageableListView(string, all, i)
@Override
protected void populateItem(ListItem item) {
try { final Product product = (Product) item.getModelObject();

```

```

String name2 = product.getName();
Label name = new Label("nameCat", new Model(name2));
AjaxFallbackLink lnkDetailCat = new AjaxFallbackLink(
    "lnkDetailCat") {
    @Override
    public void onClick(AjaxRequestTarget target) { } };
final PageParameters pageParameters = new PageParameters();
pageParameters.add("page", "edit");
pageParameters.add("id", product.getId().toString());
AjaxFallbackLink moreDetail = new AjaxFallbackLink(
    "moreDetail") {
    @Override
    public void onClick(AjaxRequestTarget target) {
        setResponsePage(Details.class, pageParameters); } };
String string2 = relatifpath + File.separator+
product.getId() + File.separator + "thumbnail" +
File.separator + ".jpg"; System.out.println(string2);
Image imgThumb = new Image("imgThumbCat",
getImg(string2));
moreDetail.add(imgThumb);
lnkDetailCat.add(name);
item.add(lnkDetailCat);
item.add(moreDetail); } catch (Exception e) {
e.printStackTrace(); } };
return listView; } }

```

#### CATEGORY.HTML

```

<wicket:extend>
<div class="center_content">
<div class="left_content">
<div class="crumb_nav">
<a wicket:id="lnkHome">beranda</a> &gt;&gt;
<span wicket:id="categoryName">kategori</span>
</div>
<div class="title">
<span class="title_icon"></span>KATEGORI</div>
<div class="new_products">
<div wicket:id="productCat">
<div class="new_prod_box">
<a href="details.html" wicket:id="lnkDetailCat"><label
wicket:id="nameCat">
product name </label> </a>
<div class="new_prod_bg">
<a href="#" wicket:id="moreDetail"></a></div>
</div></div>
<div class="pagination">
<span class="disabled"></span><span class="current"
wicket:id="paginator"/></div></div>
<div class="clear"></div></div>
<div class="right_content">
<div class="cart" wicket:id="cart">
<div class="title">
<span class="title_icon"></span>pesanan</div>
<div class="home_cart_content">
<label wicket:id="numberItems">3 x Items</label>
<span class="red" wicket:id="totalItems">TOTAL: 100$</span>
</div>

```

```

<a href="#" class="view_cart" wicket:id="viewCart">view
cart</a></div>
<div class="title">
<span class="title_icon"></span>aurora komputer</div>
<div class="about">
<p>
Aurora.com menjual peralatan komputer, terutama
lebih terkait ke Personal Komputer, Projector, Baterai, Adaptor,
Accessories dan lainnya.</p></div>

```

```

<div class="right_box">
<div class="title">
<span class="title_icon"></span>KATEGORI</div>
<ul class="list" wicket:id="categories">
<li><a wicket:id="lnkuser" style="cursor: pointer"> <label
wicket:id="lblUsername">[categoryname]</label> </a></li></ul>
</div></div><!--end of right content-->
<div class="clear"></div></div><!--end of center content-->
</wicket:extend>

```

#### CHECKOUT.JAVA

```

package com.roes.commerce.pages.frontpage;
import java.util.ArrayList;
public class Checkout extends FrontPage{
    public WebMarkupContainer cart = new
    WebMarkupContainer("cart");
    private DropDownChoice countryDDC;
    private DropDownChoice zoneDDC;
    public Checkout() throws Exception{
        PageableListView categories = createListCategories("categories",
        serviceCRUDProduct.getAllCategory(), 10);
        add(categories);
        cart.add(new Label("numberItems", new
        Model(getNumberItems())));
        cart.add(new Label("totalItems", new Model(getTotalItems())));
        cart.add(new AjaxFallbackLink("viewCart")){
            @Override
            public void onClick(AjaxRequestTarget target) {
                setResponsePage(Cart.class); } };
        add(cart); add(createForm("form")); }
    private Form createForm(String id){
        Form form = new Form(id);
        IChoiceRenderer iChoiceCountry = new IChoiceRenderer() {
            private static final long serialVersionUID = 1L;
            @Override
            public Object getDisplayValue(Object obj) {
                Country w = (Country) obj; return w.getName(); }
            @Override
            public String getIdValue(Object obj, int indeks) {
                Country w = (Country) obj; return w.getId().toString();
            } };
        IChoiceRenderer iChoiceZone = new IChoiceRenderer() {
            private static final long serialVersionUID = 1L;
            @Override
            public Object getDisplayValue(Object obj) {
                Zone w = (Zone) obj; return w.getName(); }
            @Override
            public String getIdValue(Object obj, int indeks) {
                Zone w = (Zone) obj; return w.getId().toString(); } };
    }
}

```

```

final TextField username = (TextField)new TextField("username",
new Model()).setRequired(false);
final TextField lastname = (TextField)new TextField("lastname",
new Model()).setRequired(false);
final TextField address1 = (TextField)new TextField("address1",
new Model()).setRequired(false);
final TextField address2 = (TextField)new TextField("address2",
new Model()).setRequired(false);
final TextField mail = (TextField)new TextField("email", new
Model()).setRequired(false);
final TextField phone = (TextField)new TextField("phone", new
Model()).setRequired(false);
final TextField company = (TextField)new TextField("company",
new Model()).setRequired(false);
final TextField fax = (TextField)new TextField("fax", new
Model());
final TextField city = (TextField)new TextField("city", new
Model()).setRequired(false);
final TextField postcode = new TextField("postcode", new
Model());
countryDDC = new DropDownChoice("country", new Model(),
serviceCRUDProduct.getAllCountry(), iChoiceCountry);
zoneDDC = (DropDownChoice) new DropDownChoice("zone",
new Model(), new ArrayList<Zone>(),
iChoiceZone).setOutputMarkupId(true);
countryDDC.add(new
AjaxFormComponentUpdatingBehavior("onchange") {
protected void onUpdate(AjaxRequestTarget target) {
String stringWhId = countryDDC.getValue().trim();
if (!stringWhId.equals("-1")) {
List<Zone> listZone =
serviceCRUDProduct.getZoneByCountry(Integer.parseInt(stringW
hId));
System.out.println("data "+listZone.size());
if (listZone.size() > 0) {
zoneDDC.setModel(new Model());
zoneDDC.setChoices(listZone);
target.addComponent(zoneDDC); } } else {
zoneDDC.setChoices(new ArrayList<Zone>());
target.addComponent(zoneDDC); } } });
final FeedbackPanel msg = (FeedbackPanel) new
FeedbackPanel("msg").setOutputMarkupId(true);
AjaxButton btnSave = new AjaxButton("btnSave"){
@Override
protected void onSubmit(AjaxRequestTarget target, Form form) {
target.addComponent(msg);
isLengthNotZero(repassword)>0){
if(isLengthNotZero(username)>0){
if(isLengthNotZero(address1)>0){
if(isLengthNotZero(mail)>0){
if(isLengthNotZero(phone)>0){
if(isLengthNotZero(city)>0){
if(isLengthNotZero(postcode)>0){
Address address = new Address();
address.setAddress1(address1.getValue());
address.setAddress2(address2.getValue());
address.setCity(city.getValue());
address.setCompany(company.getValue());
Country country =
serviceCRUDProduct.getCountryById(Integer.parseInt(countryDD
C.getValue()));

```

```

address.setCountry(country);
address.setFirstname(username.getValue());
address.setLastname(lastname.getValue());
address.setPostcode(postcode.getValue());
Zone zone =
serviceCRUDProduct.getZoneById(Integer.parseInt(zoneDDC.get
Value()));
address.setZone(zone);
Customer customer = new Customer();
customer.setAddress(address);
customer.setDateAdded(new Date());
customer.setEmail(mail.getValue());
customer.setFax(fax.getValue());
customer.setFirstname(username.getValue());
customer.setLastname(lastname.getValue());
customer.setPassword("sss");
customer.setStatus(new Integer(0));
customer.setTelephone(phone.getValue());
Map<Product, Integer> mapProduct = ((WicketSession)
getSession()).getMapProduct();
Set<Product> setProduct = ((WicketSession)
getSession()).getSetProduct();
Orders orders = serviceCRUDProduct.saveOrders(address,
customer, mapProduct, setProduct);
List<OrdersProduct> listOrderProduct =
getListOrderProduct(mapProduct, setProduct, orders);
serviceCRUDProduct.saveOrderDetail(listOrderProduct);
final PageParameters pageParameters = new PageParameters();
pageParameters.add("page", "edit");
pageParameters.add("id", orders.getOrderId().toString());
((WicketSession) getSession()).setProducts(new
ArrayList<Product>());
((WicketSession) getSession()).setMapProduct(new
HashMap<Product, Integer>());
((WicketSession) getSession()).setSetProduct(new
HashSet<Product>());
setResponsePage(Confirm.class, pageParameters); }else{
giveAlert(target, "postcode"); }
}else{ giveAlert(target, "city"); }
}else { giveAlert(target, "phone"); }
}else{ giveAlert(target, "mail"); }
}else{ giveAlert(target, "address1"); }
}else{ giveAlert(target, "username"); } }
private List<OrdersProduct> getListOrderProduct(
Map<Product, Integer> mapProduct, Set<Product> setProduct,
Orders orders) {List<OrdersProduct> list = new
ArrayList<OrdersProduct>();
for (Product product : setProduct) {
Integer sum = mapProduct.get(product);
Double totalPrice = sum * product.getPrice();
OrdersProduct ordersProduct = new OrdersProduct();
ordersProduct.setName(product.getName());
ordersProduct.setOrders(orders);
ordersProduct.setPrice(product.getPrice());
ordersProduct.setProduct(product);
ordersProduct.setQuantity(sum);
ordersProduct.setTotal(totalPrice);
list.add(ordersProduct); }
return list; }
private void giveAlert(AjaxRequestTarget target, String name) {

```



```

target.appendJavascript("alert('data field "+name+" masih
kosong)"); }
private int isLengthNotZero(final TextField password) {
return password.getValue().length(); } };
form.add(username);
form.add(address1);
form.add(address2);
form.add(mail);
form.add(phone);
form.add(company);
form.add(fax);
form.add(city);
form.add(countryDDC);
form.add(zoneDDC);
form.add(btnSave);
form.add(msg);
form.add(postcode);
form.add(lastname);
return form; } }

```

### CHECKOUT.HTML

```

<wicket:extend>
<div class="center_content">
<div class="left_content">
<div class="title"><span class="title_icon"></span>Register</div>
<div class="feat_prod_box_details">
<p class="details">
Isi data data di form di bawah dengan benar. Kebenaran
data yang anda kirim akan memberikan ketepatan dan kelancaran
proses transaksi belanja anda. Sekian Terima Kasih. </p>
<div class="contact_form">
<div class="form_subtitle">create new account</div>
<form name="register" action="#" wicket:id="form">
<span wicket:id="msg"></span>
<div class="form_row">
<label class="contact"><strong>NAMA:</strong></label>
<input type="text" class="contact_input" wicket:id="username"/>
</div>
<div class="form_row">
<label class="contact"><strong>NAMA
LENGKAP:</strong></label>
<input type="text" class="contact_input" wicket:id="lastname"/>
</div>
<div class="form_row">
<label class="contact"><strong>EMAIL:</strong></label>
<input type="text" class="contact_input" wicket:id="email"/>
</div>
<div class="form_row">
<label class="contact"><strong>TELEPON:</strong></label>
<input type="text" class="contact_input" wicket:id="phone"/>
</div>
<div class="form_row">
<label
class="contact"><strong>PERUSAHAAN:</strong></label>
<input type="text" class="contact_input" wicket:id="company"/>
</div>
<div class="form_row">
<label class="contact"><strong>NEGARA :</strong></label>
<select class="contact_input"
wicket:id="country"></select></div>

```

```

<div class="form_row">
<label class="contact"><strong>PROPINSI :</strong></label>
<select class="contact_input" wicket:id="zone"></select>
</div>
<div class="form_row">
<label class="contact"><strong>FAX:</strong></label>
<input type="text" class="contact_input" wicket:id="fax"/>
</div>
<div class="form_row">
<label class="contact"><strong>KOTA:</strong></label>
<input type="text" class="contact_input" wicket:id="city"/>
</div>
<div class="form_row">
<label class="contact"><strong>ALAMAT
RUMAH:</strong></label>
<input type="text" class="contact_input" wicket:id="address1"/>
</div>
<div class="form_row">
<label class="contact"><strong>KODE POS:</strong></label>
<input type="text" class="contact_input" wicket:id="postcode"/>
</div>
<div class="form_row">
<label class="contact"><strong>ALAMAT
KANTOR:</strong></label>
<input type="text" class="contact_input" wicket:id="address2"/>
</div>
<div class="form_row">
<div class="terms"><input type="checkbox" name="terms" />
I agree to the <a href="#">terms & conditions</a>
</div></div>
<div class="form_row">
<input type="submit" class="register" value="register"
wicket:id="btnSave"/></div> </form></div></div>
<div class="clear"></div>
</div><!--end of left content-->
<div class="right_content">
<div class="cart" wicket:id="cart">
<div class="title">
<span class="title_icon"> </span>pesanan</div>
<div class="home_cart_content">
<label wicket:id="numberItems">3 x Items</label>
<span class="red" wicket:id="totalItems">TOTAL: 100$</span>
</div><a href="#" class="view_cart" wicket:id="viewCart">view
cart</a></div>
<div class="title">
<span class="title_icon"> </span>aurora komputer</div>
<div class="about">
<p>
Aurora.com menjual peralatan komputer, terutama
lebih terkait ke Personal Komputer, Projector, Baterai, Adaptor,
Accessories dan lainnya.</p></div>
<div class="right_box">
<div class="title">
<span class="title_icon"> </span>KATEGORI</div>
<ul class="list" wicket:id="categories">
<li><a wicket:id="lnkuser" style="cursor: pointer"> <label
wicket:id="lblUsername">[clientusername]
</label> </a></li></ul></div></div>

```

```

<!--end of right content-->
<div class="clear"></div></div>
<!--end of center content-->
</wicket:extend>

```

### CONFIRM.JAVA

```

package com.roes.commerce.pages.frontpage;
import java.util.Collection;
public class Confirm extends FrontPage {
public WebMarkupContainer cart = (WebMarkupContainer) new
WebMarkupContainer("cart").setOutputMarkupId(true);
private Map<Product, Integer> mapProducts = new
HashMap<Product, Integer>();
private double totalPrice = 0.0;
private Set<Product> setProduct = new HashSet<Product>();
public Confirm(PageParameters pageParameters) throws
Exception{
PageableView categories = createListCategories("categories",
serviceCRUDProduct.getAllCategory(), 10);
add(categories);
cart.add(new Label("numberItems", new
Model(getNumberItems())));
cart.add(new Label("totalItems", new Model(getTotalItems())));
cart.add(new AjaxFallbackLink("viewCart") {
@Override
public void onClick(AjaxRequestTarget target) {
setResponsePage(Cart.class); } });
add(cart);
add(new PageLink("continue", Index.class));
Integer id = Integer.parseInt(pageParameters.getString("id"));
Orders orders = serviceCRUDProduct.getOrdersById(id);
Customer customer = orders.getCustomer();
Zone zone = customer.getAddress().getZone();
TaxRate taxRate =
serviceCRUDProduct.getTaxRateByZone(zone);
System.out.println("zone id " + zone.getZoneId());
Collection<OrdersProduct> ordersProducts =
orders.getOrdersProduct();
double totalPrice = 0.0;
for (OrdersProduct ordersProduct : ordersProducts) {
totalPrice+= ordersProduct.getTotal(); }
add(new Label("customerName", new
Model(customer.getFirstname())));
add(new Label("total", new Model(totalPrice)));
add(new Label("rekening", new Model(Constants.REKENING)));
} }

```

### CONFIRM.HTML

```

<wicket:extend>
<div class="center_content">
<div class="left_content">
<div class="title">
<span class="title_icon">
</span>pesanan
</div>
<div class="feat_prod_box_details">
Selamat saudara <span style="font-weight: bold;color:
blue" wicket:id="customerName"></span> anda berhasil
melakukan transaksi, Jumlah yang harus anda bayar adalah <span
style="font-weight: bold;color: blue" wicket:id="total"></span>
plus biaya kirim dapat dilihat di <a

```

```

href="http://www.jne.co.id/">jne.co.id</a>.Silahkan kirim ke
Rekening berikut <span style="font-weight: bold;color: blue"
wicket:id="rekening"></span>.
<p>Sekian terima kasih atas transaksi anda.</p>
<a href="#" class="continue" wicket:id="continue">&lt;
continue</a></div><div class="clear"></div></div>
<!--end of left content-->
<div class="right_content"><div class="cart" wicket:id="cart">
<div class="title">
<span class="title_icon"></span>My cart</div>
<div class="home_cart_content">
<label wicket:id="numberItems">3 x Items</label>
<span class="red" wicket:id="totalItems">TOTAL: 100$</span>
</div>
<a href="#" class="view_cart" wicket:id="viewCart">view
cart</a></div>
<div class="title">
<span class="title_icon"></span>About Our Store</div>
<div class="about">
<p>
Aurora.com menjual peralatan komputer, terutama
lebih terkait ke Personal Komputer, Projector, Baterai, Adaptor,
Accesorries dan lainnya.</p> </div>
<div class="right_box">
<div class="title">
<span class="title_icon"></span>Categories</div>
<ul class="list" wicket:id="categories">
<li><a wicket:id="Inkuser" style="cursor: pointer"> <label
wicket:id="lblUsername">[categoryname]</label> </a></li></ul>
</div></div><div class="clear"></div></div></wicket:extend>

```

### CONTACT.JAVA

```

package com.roes.commerce.pages.frontpage;
import java.util.List;
public class contact extends FrontPage {
public contact() throws Exception{
PageableView categories = createListCategories("categories",
serviceCRUDProduct.getAllCategory(), 10);
add(categories); }
}

```

### CONTACT.HTML

```

<wicket:extend>
<div class="center_content">
<div class="left_content">
<div class="title">
<span class="title_icon"></span>TENTANG KAMI</div>
<div class="feat_prod_box_details">
<p class="details">

Contact langsung pada about
</p></div>
<div class="clear"></div></div>
<div class="right_content">
<div class="cart">
<div class="title"><span class="title_icon"></span>My cart</div>

```



Aurora.com menjual peralatan komputer, terutama lebih terkait ke Personal Komputer, Projector, Baterai, Adaptor, Accesories dan lainnya. </p> </div>  
<div class="right\_box">  
<div class="title">  
<span class="title\_icon"> </span>KATEGORI</div>  
<ul class="list" wicket:id="categories">  
<li><a wicket:id="Inkuser" style="cursor: pointer"> <label wicket:id="lblUsername">[categoryname]</label> </a></li>  
</ul></div></div>  
<!--end of right content-->  
<div class="clear"></div></div><!--end of center content-->  
</wicket:extend>

## FRONTPAGE.JAVA

```
package com.roes.commerce.pages.frontpage;
import java.awt.image.BufferedImage;
public class FrontPage extends WebPage {
    private PageLink home;
    private PageLink about;
    private PageLink register;
    private PageLink products;
    private PageLink specials;
    private PageLink details;
    private PageLink contact;
    private PageLink account;
    private PageLink homef;
    private PageLink aboutf;
    private PageLink contactf;
    public List<Product> listProducts = ((WicketSession)
getSession()).getProducts();
    public ServiceCRUDProduct serviceCRUDProduct =
ServiceCRUDProduct.getInstance();
    public String path = System.getProperty("catalina.base") +
File.separator + "webapps" + File.separator +
Constants.projectName+ File.separator + "images";
    public String relatifpath = "images";
    public FrontPage() {
        home = createPageLink("home", Index.class);
        about = createPageLink("about", About.class);
        products = createPageLink("products", Index.class);
        details = createPageLink("details", Login.class);
        contact = createPageLink("contact", contact.class);
        homef = createPageLink("homef", Index.class);
        aboutf = createPageLink("aboutf", About.class);
        contactf = createPageLink("contactf", Index.class);
        add(home);
        add(about);
        add(products);
        add(details);
        add(contact);
        add(homef);
        add(aboutf);
        add(contactf); }
    private PageLink createPageLink(String id, Class classPage) {
        PageLink pl = new PageLink(id, classPage);
        return pl; }
    public void addLabelToItem(Label[] label, ListItem item) {
        for (Label label2 : label) {item.add(label2); } }
```

```
public BufferedDynamicImageResource getImage(String
imagePath)
throws IOException {BufferedDynamicImageResource resource =
new BufferedDynamicImageResource();
    BufferedImage image = null;
    try { image = ImageIO.read(((WebApplication) Application.get())
.getServletContext().getResourceAsStream(imagePath));
    } catch (IOException ioe)
    {
        image = ImageIO.read(((WebApplication) Application.get())
.getServletContext().getResourceAsStream("images" +
File.separator + "no_image.jpg"));
        ioe.printStackTrace(); }
        resource.setImage(image);
        return resource; }
    public String getNumberItems(){
        Integer items = 0;
    return listProducts.size() + Constants.itemsCount; }
    public String getTotalItems(){
        Double sumPrice = 0.0;
        for (Product product : listProducts) {
            sumPrice+=product.getPrice(); }
        return sumPrice.toString(); }
    public PageableListView createListCategories(String string,
List<com.roes.commerce.model.Category> all, int i) throws
Exception {
        PageableListView listView = new PageableListView(string, all, i)
        {
            @Override
            protected void populateItem(ListItem item) {
                try { final com.roes.commerce.model.Category category =
(com.roes.commerce.model.Category) item.getModelObject();
                String name2 = category.getCategoryDescription().getName();
                AjaxFallbackLink Inkuser = new AjaxFallbackLink("Inkuser") {
                    @Override
                    public void onClick(AjaxRequestTarget target) {
                        PageParameters parameters = new PageParameters();
                        parameters.add("page", "edit");
                        parameters.add("id", category.getCategoryId().toString());
                        setResponsePage(com.roes.commerce.pages.frontpage.Category.cl
ass.parameters); } } };
                Label lblUsername = new Label("lblUsername", new Model(
name2));
                Inkuser.add(lblUsername);
                item.add(Inkuser); } catch (Exception e) {
                    e.printStackTrace(); } } };
        return listView; } }
```

## FRONTPAGE.HTML

```
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=windows-1252" />
<title>Best Online IT Shop. Roes Corp</title>
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
<div id="wrap">
<div class="header">
<div class="logo">
<a href="index.html"></a></div>
```

```

<div id="menu" >
<ul><li class="selected">
<a wicket:id="home">BERANDA</a></li>
<li><a wicket:id="about">TENTANG KAMI</a></li>
<li><a wicket:id="products"> PRODUK</a></li>
<li><a wicket:id="details">ADMIN</a></li>
<li><a
wicket:id="contact">KONTAK</a></li></ul></div></div>
<wicket:child></wicket:child>
<div class="footer">
<div class="left_footer">
<br />
<a href="http://csscreme.com/freecsstemplates/"title="free
templates"></a></div>
<div class="right_footer">
<a wicket:id="homef">beranda</a>
<a wicket:id="aboutf">tentang kami</a>
<a href="#">layanana</a>
<a href="#">privasi</a>
<a wicket:id="contactf">kontak</a></div></div></div>
</body>
</html>

```

#### INDEX.JAVA

```

package com.roes.commerce.pages.frontpage;
import java.io.File;
public class Index extends FrontPage {
public WebMarkupContainer cart = new
WebMarkupContainer("cart");
public Index() throws Exception {
List<Product> all = serviceCRUDProduct.getAll();
PageableView dataView = createTableData("productx", all, 4);
PageableView categories = createListCategories("categories"
serviceCRUDProduct.getAllCategory(), 10);
PageableView newProducts =
createListNewProduct("newProducts",
serviceCRUDProduct.getNewestProducts(), 3);
add(dataView);
add(categories);
add(newProducts);
cart.add(new Label("numberItems", new
Model(getNumberItems())));
cart.add(new Label("totalItems", new
Model(getTotalItems())));
cart.add(new AjaxFallbackLink("viewCart"){
@Override
public void onClick(AjaxRequestTarget target) {
setResponsePage(Cart.class); } }); add(cart); }
private PageableView createTableData(String string,
List<Product> all, int i) throws Exception {
PageableView listView = new PageableView(string, all, i)
{
@Override
protected void populateItem(ListItem item) {
try { final Product product = (Product) item.getModelObject();
String name2 = product.getName();
Label name = new Label("prodname", new Model(name2));
Label desc = new Label("prodDesc", new
Model(product.getDescs()));

```

```

String string2 = relatifpath + File.separator+
product.getProductId() + File.separator+ "thumbnail" +
File.separator + "1.jpg";
System.out.println(string2);
Image imgThunb = new Image("imgThunb", getImage(string2));
final PageParameters pageParameters = new PageParameters();
pageParameters.add("page", "edit");pageParameters.add("id",
product.getProductId().toString());
AjaxFallbackLink lnkDetail = new AjaxFallbackLink(
"lnkDetail") {
@Override
public void onClick(AjaxRequestTarget target) {
setResponsePage(Details.class, pageParameters); }
}AjaxFallbackLink moreDetail = new AjaxFallbackLink(
"moreDetail") {
@Override
public void onClick(AjaxRequestTarget target) {
setResponsePage(Details.class, pageParameters); } };
lnkDetail.add(imgThunb);
item.add(moreDetail);
Label label[] = { name, desc };
addLabelToItem(label, item);
item.add(lnkDetail);
} catch (Exception e) {
e.printStackTrace(); } } };
return listView; }
private PageableView createListNewProduct(String string,
List<Product> all, int i) throws Exception {
PageableView listView = new PageableView(string, all, i)
{
@Override
protected void populateItem(ListItem item) {
try {
final Product product = (Product) item.getModelObject();
String name2 = product.getName();
Label name = new Label("productNameNew", new
Model(name2));
String string2 = relatifpath + File.separator+
product.getProductId() + File.separator+ "thumbnail" +
File.separator + "1.jpg";
Image imgThunb = new Image("imgThunbNew",
getImage(string2));
final PageParameters pageParameters = new PageParameters();
pageParameters.add("page", "edit");
pageParameters.add("id", product.getProductId().toString());
AjaxFallbackLink lnkDetail = new AjaxFallbackLink(
"lnkDetailNew") {
@Override
public void onClick(AjaxRequestTarget target) {
setResponsePage(Details.class, pageParameters); } };
lnkDetail.add(imgThunb);
item.add(name);
item.add(lnkDetail); } catch (Exception e) {
e.printStackTrace(); } } };
return listView; } }

```

#### INDEX.HTML

```

<wicket:extend>
<div class="center_content" style="clear: both">
<div class="left_content">
<div class="title">

```

```

<span class="title_icon"></span>FITUR PRODUK</div>
<div wicket:id="productx">
<div class="feat_prod_box">
<div class="prod_img">
<a wicket:id="lnkDetail"> </a></div>
<div class="prod_det_box">
<div class="box_top"></div>
<div class="box_center">
<div class="prod_title" wicket:id="prodname">Product
name</div>
<p class="details" wicket:id="prodDesc">Product Desc</p>
<a href="#" class="more" wicket:id="moreDetail"> detail produk
</a>
<div class="clear"></div></div>
<div class="box_bottom"></div></div>
<div class="clear"></div></div></div>
<div class="title">
<span class="title_icon"> </span>PRODUK TERBARU</div>
<div class="new_products">
<div wicket:id="newProducts">
<div class="new_prod_box">
<a href="details.html"><label wicket:id="productNameNew">
product name</label> </a>
<div class="new_prod_bg">
<span class="new_icon"> </span>
<a href="#"
wicket:id="lnkDetailNew"><imgsrc="images/thumb1.gif" alt=""
title="" class="thumb" border="0" wicket:id="imgThumbNew" />
</a></div></div></div></div>

```

```

<div class="clear"></div></div><!--end of left content-->
<div class="right_content">
<div class="cart" wicket:id="cart">
<div class="title">
<span class="title_icon"> </span>My cart</div>
<div class="home_cart_content">
<label wicket:id="numberItems">3 x Items</label>
<span class="red" wicket:id="totalItems">TOTAL:
100$</span></div>
<a href="#" class="view_cart"
wicket:id="viewCart">pesanan</a></div>
<div class="title">
<span class="title_icon"> </span>aurora komputer</div>
<div class="about">
<p>
Aurora.com menjual peralatan komputer, terutama
lebih terkait ke Personal Komputer, Projector, Baterai, Adaptor,
Accesorries dan lainnya.</p></div>
<div class="right_box">
<div class="title">
<span class="title_icon"> </span>KATEGORI</div>
<ul class="list" wicket:id="categories">
<li><a wicket:id="lnkuser" style="cursor: pointer"> <label
wicket:id="lblUsername">[clientusername]</label> </a>
</li></ul></div></div><!--end of right content-->
<div class="clear"></div></div>
<!--end of center content-->
</wicket:extend>

```