**DAFTAR LAMPIRAN**

**SERVER SIDE SOURCE CODE**

1. **Source Code Config Database :**

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

$active\_group = 'erph';

$active\_record = TRUE;

$db['erph']['hostname'] = 'localhost';

$db['erph']['username'] = 'postgres';

$db['erph']['password'] = 'dbadmin';

$db['erph']['database'] = 'erp';

$db['erph']['dbdriver'] = 'postgre';

$db['erph']['dbprefix'] = '';

$db['erph']['pconnect'] = TRUE;

$db['erph']['db\_debug'] = TRUE;

$db['erph']['cache\_on'] = FALSE;

$db['erph']['cachedir'] = '';

$db['erph']['char\_set'] = 'utf8';

$db['erph']['dbcollat'] = 'utf8\_general\_ci';

$db['erph']['swap\_pre'] = '';

$db['erph']['autoinit'] = TRUE;

$db['erph']['stricton'] = FALSE;

/\*\*

\* hrd

\*\*/

$db['hrd']['hostname'] = 'localhost';

$db['hrd']['username'] = 'hrdusr';

$db['hrd']['password'] = 'hrdpswd';

$db['hrd']['database'] = 'dbhrd';

$db['hrd']['dbdriver'] = 'postgre';

$db['hrd']['dbprefix'] = '';

$db['hrd']['pconnect'] = TRUE;

$db['hrd']['db\_debug'] = TRUE;

$db['hrd']['cache\_on'] = FALSE;

$db['hrd']['cachedir'] = '';

$db['hrd']['char\_set'] = 'utf8';

$db['hrd']['dbcollat'] = 'utf8\_general\_ci';

$db['hrd']['swap\_pre'] = '';

$db['hrd']['autoinit'] = TRUE;

$db['hrd']['stricton'] = FALSE;

/\* End of file database.php \*/

/\* Location: ./application/config/database.php \*/

1. **Source Code Config :**

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

$config['base\_url'] = ((isset($\_SERVER['HTTPS']) && $\_SERVER['HTTPS'] == "on") ? "https" : "http");

$config['base\_url'] .= "://".$\_SERVER['HTTP\_HOST'];

$config['base\_url'] .= str\_replace(basename($\_SERVER['SCRIPT\_NAME']),"",$\_SERVER['SCRIPT\_NAME']);

$config['index\_page'] = '';

$config['uri\_protocol'] = 'AUTO';

$config['url\_suffix'] = '';

$config['language'] = 'english';

$config['charset'] = 'UTF-8';

$config['enable\_hooks'] = FALSE;

$config['subclass\_prefix'] = 'MY\_';

$config['permitted\_uri\_chars'] = 'a-z 0-9~%.:\_\-';

$config['allow\_get\_array'] = TRUE;

$config['enable\_query\_strings'] = FALSE;

$config['controller\_trigger'] = 'c';

$config['function\_trigger'] = 'm';

$config['directory\_trigger'] = 'd'; // experimental not currently in use

$config['log\_threshold'] = 0;

$config['log\_path'] = '';

$config['log\_date\_format'] = 'Y-m-d H:i:s';

$config['cache\_path'] = '';

$config['encryption\_key'] = '123';

$config['sess\_cookie\_name'] = 'ci\_session';

$config['sess\_expiration'] = 7200;

$config['sess\_expire\_on\_close'] = FALSE;

$config['sess\_encrypt\_cookie'] = FALSE;

$config['sess\_use\_database'] = FALSE;

$config['sess\_table\_name'] = 'ci\_sessions';

$config['sess\_match\_ip'] = FALSE;

$config['sess\_match\_useragent'] = TRUE;

$config['sess\_time\_to\_update'] = 300;

$config['cookie\_prefix'] = "";

$config['cookie\_domain'] = "";

$config['cookie\_path'] = "/";

$config['cookie\_secure'] = FALSE;

$config['global\_xss\_filtering'] = FALSE;

$config['csrf\_protection'] = FALSE;

$config['csrf\_token\_name'] = 'csrf\_test\_name';

$config['csrf\_cookie\_name'] = 'csrf\_cookie\_name';

$config['csrf\_expire'] = 7200;

$config['compress\_output'] = FALSE;

$config['time\_reference'] = 'local';

$config['rewrite\_short\_tags'] = FALSE;

$config['proxy\_ips'] = '';

1. **Source Code Main Controller :**

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Main extends CI\_Controller

{

function \_\_construct()

{

parent::\_\_construct();

}

function index()

{

$this->load->view('authlogin');

}

function validasi\_credential()

{

$this->form\_validation->set\_rules('username','Username','trim|required|xss\_clean');

$this->form\_validation->set\_rules('password','Password','trim|required|xss\_clean|callback\_check\_database');

if($this->form\_validation->run() == FALSE)

{

$this->index();

}

else

{

redirect('site/user\_area/');

}

}

function check\_database()

{

$this->load->model('user\_model');

$username = $this->input->post('username', TRUE);

$password = $this->input->post('password');

$result = $this->user\_model->validasi\_user($username, $password); //Proses Parsing Variabel Ke Model

// var\_dump($result);

// exit();

if($result)

{

$sess\_array = array();// Deklarasi Variabel Yg Bersifat Array

foreach($result as $row)

{

$sess\_array = array('logged\_in' => true, 'username' => $row->username, 'id' => $row->ad\_user\_id); // Array Definisi yg bersifat true

$this->session->set\_userdata($sess\_array);

}

// var\_dump($sess\_array);

// exit();

return TRUE;

}

else

{

$this->form\_validation->set\_message('check\_database', 'Invalid Username or Password');

return FALSE;

}

}

function logout()

{

$this->session->sess\_destroy();

$this->index();

}

}

1. **Source Code Site Contrller :**

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Site extends CI\_Controller

{

function \_\_construct()

{

parent::\_\_construct();

}

function user\_area()

{

if($this->session->userdata('logged\_in') == true)

{

$this->load->model('user\_model');

$iscreate = $this->user\_model->iscreatePrevilege()->result();

$isupdate = $this->user\_model->isupdatePrevilege()->result();

$isdelete = $this->user\_model->isdeletePrevilege()->result();

$isprocess = $this->user\_model->isprocessPrevilege()->result();

$previlege = '';

foreach($iscreate as $create){

$previlege .= 'var create'.$create->menu.' = '.$create->iscreate.'; ';

}

foreach($isupdate as $update){

$previlege .= 'var update'.$update->menu.' = '.$update->isupdate.'; ';

}

foreach($isdelete as $delete){

$previlege .= 'var delete'.$delete->menu.' = '.$delete->isdelete.'; ';

}

foreach($isprocess as $process){

$previlege .= 'var process'.$process->menu.' = '.$process->isprocess.'; ';

}

$data['previlege'] = $previlege;

$this->load->view('dashboard', $data);

}

else

{

$this->load->view('authlogin');

}

}

}

1. **Source Code User Model :**

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class User\_model extends CI\_Model

{

private $connectionName;

public function \_\_construct(){

parent::\_\_construct();

}

public function setConnection($connectionName){

$this->connectionName = $connectionName;

}

public function getConnection(){

return $this->load->database($this->connectionName,TRUE);

}

function validasi\_user($username, $password)

{

$this->setConnection('erph');

$db = $this->getConnection();

$db->select('ad\_user\_id, username, password');

$db->from('ad\_user');

$db->where('username',$username);

$db->where('password',base64\_encode(sha1($password,TRUE)));

$db->where('isactive','Y');

$db->limit(1);

$query = $db->get();

if($query->num\_rows() == 1)

{

return $query->result();

}

else

{

return FALSE;

}

}

function user\_rules($id)

{

$this->setConnection('erph');

$db = $this->getConnection();

$db->select('gm.ad\_menu\_id AS id');

$db->from('ad\_role\_menu AS gm');

$db->join('ad\_role r', 'r.ad\_role\_id=gm.ad\_role\_id');

$db->join('ad\_user u', 'r.ad\_role\_id=u.ad\_role\_id');

$db->where('u.ad\_user\_id',$id);

$query = $db->get();

return $query;

}

public function cekrulesparent($rules){

$this->setConnection('erph');

$db = $this->getConnection();

$db->select('ad\_menu\_id AS id, name AS name, icon AS icon, leaf AS leaf, selector AS selector, cls AS cls');

$db->from('ad\_menu');

$db->where\_in('ad\_menu\_id',$rules);

$db->where('parent',0);

$db->where('isactive','Y');

$db->order\_by('ad\_menu\_id');

$query = $db->get();

return $query;

}

public function cekruleschild($id, $rules){

$this->setConnection('erph');

$db = $this->getConnection();

$db->select('ad\_menu\_id AS id, name AS name, icon AS icon, leaf AS leaf, selector AS selector, cls AS cls');

$db->from('ad\_menu');

$db->where\_in('parent',$id);

$db->where('parent !=',0);

$db->where('isactive','Y');

$db->where\_in('ad\_menu\_id',$rules);

$db->order\_by('ad\_menu\_id');

$query = $db->get();

return $query;

}

function validasi\_rule($node)

{

$menus = $this->user\_rules($this->session->userdata('id'));

foreach($menus->result() as $val){

$data[] = $val->id;

}

// var\_dump($data);

// exit();

if ($node) {

$result = $this->cekruleschild($node, $data);

} else {

$result = $this->cekrulesparent($data);

}

foreach ($result->result() as $key => $value) {

$rules[] = array(

'id' => $value->id,

'text' => $value->name,

'iconCls' => $value->icon,

'leaf' => ($value->leaf === 't') ? true : false,

'selector' => $value->selector,

'cls' => $value->cls

);

}

return $rules;

}

public function iscreatePrevilege(){

$this->setConnection('erph');

$db = $this->getConnection();

$db->select(" REPLACE(m.name, ' ', '') AS menu, CASE WHEN rm.iscreate = 'Y' THEN 'false' ELSE 'true' END AS iscreate", FALSE);

$db->from('ad\_role\_menu rm');

$db->join('ad\_role r','r.ad\_role\_id=rm.ad\_role\_id');

$db->join('ad\_user u','r.ad\_role\_id=u.ad\_role\_id');

$db->join('ad\_menu m','rm.ad\_menu\_id=m.ad\_menu\_id');

$db->where('u.ad\_user\_id',$this->session->userdata('id'));

$query = $db->get();

return $query;

}

public function isupdatePrevilege(){

$this->setConnection('erph');

$db = $this->getConnection();

$db->select(" REPLACE(m.name, ' ', '') AS menu, CASE WHEN rm.isupdate = 'Y' THEN 'false' ELSE 'true' END AS isupdate", FALSE);

$db->from('ad\_role\_menu rm');

$db->join('ad\_role r','r.ad\_role\_id=rm.ad\_role\_id');

$db->join('ad\_user u','r.ad\_role\_id=u.ad\_role\_id');

$db->join('ad\_menu m','rm.ad\_menu\_id=m.ad\_menu\_id');

$db->where('u.ad\_user\_id',$this->session->userdata('id'));

$query = $db->get();

return $query;

}

public function isdeletePrevilege(){

$this->setConnection('erph');

$db = $this->getConnection();

$db->select(" REPLACE(m.name, ' ', '') AS menu, CASE WHEN rm.isdelete = 'Y' THEN 'false' ELSE 'true' END AS isdelete", FALSE);

$db->from('ad\_role\_menu rm');

$db->join('ad\_role r','r.ad\_role\_id=rm.ad\_role\_id');

$db->join('ad\_user u','r.ad\_role\_id=u.ad\_role\_id');

$db->join('ad\_menu m','rm.ad\_menu\_id=m.ad\_menu\_id');

$db->where('u.ad\_user\_id',$this->session->userdata('id'));

$query = $db->get();

return $query;

}

public function isprocessPrevilege(){

$this->setConnection('erph');

$db = $this->getConnection();

$db->select(" REPLACE(m.name, ' ', '') AS menu, CASE WHEN rm.isprocess = 'Y' THEN 'false' ELSE 'true' END AS isprocess", FALSE);

$db->from('ad\_role\_menu rm');

$db->join('ad\_role r','r.ad\_role\_id=rm.ad\_role\_id');

$db->join('ad\_user u','r.ad\_role\_id=u.ad\_role\_id');

$db->join('ad\_menu m','rm.ad\_menu\_id=m.ad\_menu\_id');

$db->where('u.ad\_user\_id',$this->session->userdata('id'));

// $this->db->where('m.name','Users');

$query = $db->get();

return $query;

}

}

**CLIENT SIDE SOURCE CODE :**

1. **Source Code Defining Class app.js :**

Ext.Loader.setConfig({

enabled : true,

paths : {

'Ext' : BASE\_URL + ROOTDIR + 'extjs/src',

'Ext.ux' : BASE\_URL + ROOTDIR + 'extjs/ux',

}

});

Ext.application({

name : 'ERPh',

appFolder : BASE\_URL + ROOTDIR + 'app',

controllers : ['ERPh.controller.dashboard'],

autoCreateViewport: true,

launch : function()

{

var me = this;

setInterval(me.jamDigital, 1000);

},

jamDigital: function()

{

var tanggal = new Date();

//Jam

var jam = tanggal.getHours().toString();

if(jam.length == 1){

jam = '0' + jam;

}

var menit = tanggal.getMinutes().toString();

if(menit.length==1){

menit = '0' + menit;

}

var detik = tanggal.getSeconds().toString();

if(detik.length==1){

detik = '0' + detik;

}

Ext.getCmp("timeStatus").update(jam + ":" + menit + ":" + detik);

Ext.getCmp("dateStatus").update(tanggal.getDate() + '-' + (tanggal.getMonth() + 1) + '-' + tanggal.getFullYear());

}

});

1. **Controller**

Ext.define('ERPh.controller.dashboard',{

extend : 'Ext.app.Controller',

init: function() {

var me = this;

me.getStore('ERPh.store.TreeStore');

me.control({

'mainmenu button[action=logout]' :{

click : me.logoutUser

},

'mainmenu' :{

itemclick: me.loadView

},

'mainmenu button[action=rubah]' :{

click: me.rubahPswd

},

"editpswd button[action=rubah]" : {

click: me.rubah

}

});

me.callParent(arguments);

},

logoutUser : function(button){

window.location = BASE\_URL + 'main/logout'

},

rubahPswd : function(button){

var win = Ext.create('ERPh.module.GeneralSetup.view.form.EditPswd');

win.show();

},

rubah : function(btn){

var me = this;

var win = btn.up('window');

var form = win.down('form').getForm();

var oldpswd = form.findField('oldpswd').getValue();

var newpswd = form.findField('newpswd').getValue();

var konfpswd = form.findField('konfpswd').getValue();

Ext.MessageBox.show({

title : 'Konfirmasi',

msg : 'Anda yakin akan merubah Password?',

buttons : Ext.Msg.YESNO,

icon : Ext.MessageBox.WARNING,

width : 450,

fn : function(btn, evtObj){

if (btn == 'yes') {

Ext.Ajax.request({

url : BASE\_URL + 'GeneralSetup/c\_users/editPswd',

method : 'POST',

params : {

oldpswd : oldpswd,

newpswd : newpswd,

konfpswd : konfpswd

},

success : function(response){

var data = Ext.JSON.decode(response.responseText);

if(data.total === 1){

Ext.MessageBox.show({

title : 'Informasi',

msg : 'Password Lama Salah',

icon : Ext.MessageBox.INFO,

buttons : Ext.MessageBox.OK

});

} else if(data.total === 2) {

Ext.MessageBox.show({

title : 'Error',

msg : 'Pengisian Password Baru dan Retype Password Tidak Sama',

icon : Ext.MessageBox.ERROR,

buttons : Ext.MessageBox.OK

});

} else {

Ext.MessageBox.show({

title : 'Informasi',

msg : 'Password Telah Dirubah',

icon : Ext.MessageBox.INFO,

buttons : Ext.MessageBox.OK

});

}

}

});

}

}

});

},

loadView : function(tree, record, item, index, e, eOpts) {

var me = this;

var tab = Ext.getCmp('mainTab');

var selector = record.data.selector;

var url = record.data.cls.replace(/\_/g,'.');

var store = url.replace(/view/,'store');

if (tab.getChildByElement(selector) === null) {

var controller = url.replace(/view/,'controller');

var cmp = Ext.getCmp(selector);

if(cmp !== undefined){//Pastikan bahwa component telah di destroy

cmp.destroy();

me.getStore(store).load();

}

Ext.syncRequire(url, function(){

me.getController(controller);

me.getStore(store);

var menu = Ext.getCmp('mainTab').add(Ext.create(url).show());

tab.setActiveTab(menu);

});

}else{

me.getStore(store).reload();

tab.setActiveTab(tab);

}

}

});

1. **Model :**

Ext.define('ERPh.model.TreeStore', {

extend : 'Ext.data.Model',

fields : [

{

name : 'id',

type : 'number'

},

{

name : 'text',

type : 'string'

},

{

name : 'iconCls',

type : 'string'

},

{

name : 'leaf',

type : 'boolean'

},

{

name : 'selector',

type : 'string'

},

{

name : 'cls',

type : 'string'

}

]

});

1. **Store :**

Ext.define('ERPh.store.TreeStore', {

extend : 'Ext.data.TreeStore',

model : 'ERPh.model.TreeStore',

requires : [

'ERPh.model.TreeStore'

],

autoLoad : true,

autoSync : false,

root : {

expanded : false

},

proxy : {

type : 'ajax',

api : {

read : BASE\_URL + 'menu'

},

actionMethods : {

read : 'POST'

},

reader : {

type : 'json',

root : 'data',

successProperty : 'success'

},

writer : {

type : 'json',

writeAllFields : true,

root : 'data',

encode : true

}

}

});

1. **Viewport**

Ext.define('ERPh.view.panel.Dashboard', {

extend :'Ext.container.Container',

alias : 'widget.paneldashboard',

id :'paneldashboard',

initComponent:function(){

var a = this;

Ext.apply(a,{

title : 'DASHBOARD',

iconCls : 'icon-house',

layout :{

type:'fit', align:'stretch'

},

xtype : 'container',

border : false,

frame : true,

items :[

{

xtype : 'form',

frame : false,

layout : 'fit',

border : false,

hideHeader : true,

bodyStyle : {

background:' #DDDDFF no-repeat center center'

},

padding :'3px',

html :'<div id="welcome">Welcome to Vinoti Human Resource Informastion System</div>'

}

]

});

a.callParent(arguments);

}

});

Ext.define('ERPh.view.Tab',{

extend : 'Ext.tab.Panel',

alias : 'widget.mainTab',

resizeTabs : true,

id : 'mainTab',

layout : 'fit',

border : false,

margins : '2px 2px 2px 0px',

bodyStyle : 'background : transparent',

closeAction : 'hide',

autoDestroy : false,

frame : false,

plain : true,

enableTabScroll : true,

defaults : {

autoScroll:true

},

items : [

{

xtype:'paneldashboard'

}

]

});

Ext.define('ERPh.view.Menu',{

extend :'Ext.tree.Panel',

alias :'widget.mainmenu',

initComponent:function(){

var a=this;Ext.apply(a,{

margins : '2px 0px 2px 2px',

title : 'MAIN MENU',

id : 'mainmenu',

iconCls : 'icon-application\_home',

cls : 'scrollarea',

store : 'ERPh.store.TreeStore',

xtype : 'panel',

autoScroll : true,

rootVisible : false,

useArrows : true,

layout : 'fit',

bodyStyle : 'padding : 0px',

disabled : false,

width : 250,

border : true,

frame : false,

collapsible : true,

split : true,

items : [],

tbar : [

{

id : 'user',

text : USERNAME,

tooltip : 'Show Profile',

iconCls : 'icon-user\_b',

action : 'rubah'

},

'->',

{

id : 'logout',

text : 'LOGOUT',

tooltip : 'Exit Application',

iconCls : 'icon-cancel',

action : 'logout'

}

]

});

a.callParent(arguments)

}

});

Ext.define('ERPh.view.Viewport',

{

extend : 'Ext.container.Viewport',

requires : 'ERPh.view.Menu',

alias : 'widget.viewport',

id : 'viewport',

initComponent : function(){

Ext.apply(this,{

layout : 'border',

xtype : 'panel',

defaults : {frame: true, border: true},

border : true,

items :[

{

xtype : 'panel',

layout : 'fit',

region : 'north',

title : 'HUMAN RESOURCE INFORMATION SYSTEM',

id : 'appHeader'

},

{

xtype : 'mainmenu',

region : 'west',

width : 200

},

{

region : 'center',

xtype : 'mainTab'

},

{

id : 'mainContainer ',

xtype : 'panel',

region : 'south',

layout : 'fit',

margins : '2px 2px 2x 2px',

bbar : [

'->',

{

text : 'Copyright &copy; 2014 VL Team ERP',

disabled : true

},

'-',

{

text : 'Vinoti Enterprise Resource Planning',

disabled : true

},

'-',

{

id : 'dateStatus',

text : '01-01-1970',

disabled : true

},

'-',

{

id : 'timeStatus',

disabled : true,

text : '00:00:00'

}

]

}

]

});

this.callParent(arguments);

}

});