**Render content view**

$view = Zend\_View();

$view->setScripPath(“path to view dir”);

$view->assign('var', $value);

$view\_content = $view->render(“phtml file in directory set on method set script path”);

**Get controller path**

$moduleName = $this->getRequest()->getModuleName();

$inst = Zend\_Controller\_Front::getInstance();

$p = $inst->getControllerDirectory($moduleName);

**Get configuration**

$config = Zend\_Config\_Ini(“path to ini file”);

or

$config = Zend\_Reigistry::get('configuration');

**Get layout**

$layout = Zend\_Layout::getMvcInstance();

**Set layout in controller**

$this->getHelper('layout')->setLayout('changelog');

**Get view from layout**

$view = $layout->getView();

**Call action in other module**

$layout = Zend\_Layout::getMvcInstance();

$view = $layout->getView();

$dataReturn = $view('method', 'controller name', 'module name', Array());

**Add style sheet in controller**

$this->view->headLink()->appendStyleSheet('path to css');

Validate email

$emailValidate = Zend\_Validate\_EmailAddress();

$emailValidate->isValid('abcdef@cccc.com');

**Send mail**

$mail = new Zend\_Mail();

$mail->setBodyText("cc", "UTF-8");

$mail->setBodyHtml("cc", "UTF-8");

$mail->setFrom("noreply@test.com", "admin");

$mail->addTo("dao.th@aaaaaaaaaa.com", "Guest");

$mail->setSubject("Test mail");

try{

$mail->send();

}catch(ErrorException $e){

print "Error";

}

**Overide layout and no render in controller**

$this->\_helper->layout()->setLayout('layout');

$this->\_helper->viewRenderer->setNoRender(true);

**Get current view script path**

$this->view->getScriptPath($this->\_helper->viewRenderer->getViewScript());

print\_r($this->view->getScriptPaths());

print $this->\_helper->layout()->getLayoutPath();

//$domain=preg\_replace("/^(.\*\.)?([^.]\*\..\*)$/", "$2", $finalDomain);

Load small template in view

$this->partial(“phtml file name”);

<?php

/\*\*

\* Zend Framework

\*

\* PHP version 5

\*

\* LICENSE

\*

\* This source file is subject to the new BSD license that is bundled

\* with this package in the file LICENSE.txt.

\* It is also available through the world-wide-web at this URL:

\* http://framework.zend.com/license/new-bsd

\* If you did not receive a copy of the license and are unable to

\* obtain it through the world-wide-web, please send an email

\* to license@zend.com so we can send you a copy immediately.

\*

\* @category Zend

\* @package NA

\* @author John <john@domain.com>

\* @copyright 2014 John Doe

\* @license http://framework.zend.com/license/new-bsd New BSD License

\* @version SVN:<1.1>

\* @link http://framework.zend.com/license/new-bsd description

\*/

/\*\*

\* IndexController

\*

\* @category Zend

\* @package Zend

\* @author John <john@domain.com>

\* @copyright 2014 John Doe

\* @license http://framework.zend.com/license/new-bsd New BSD License

\* @version Release:<1.1>

\* @link http://framework.zend.com/license/new-bsd description

\* @see Zend

\*/

class TestController extends SMCore\_Controller\_Action

{

/\*\*

\* Default action

\*

\* @return void NA

\*/

public function indexAction()

{

//print\_r($this->view->getScriptPaths());

$modName = $this->getRequest()->getModuleName();

$instance = Zend\_Controller\_Front::getInstance();

$modDir = $instance->getControllerDirectory($modName);

$viewPath = realpath($modDir."/../views/mails/");

$view = new Zend\_View();

$view->setScriptPath($viewPath);

$view->assign("title", "Novabox");

$view->assign("mailto", "webmaster@test.com");

$view->assign("url", "http://novabox.com/abcdef");

$view->assign("button\_text", "Login");

$bodyHtml = $view->render("template2.phtml");

$content = "cccc";

$mail = new Zend\_Mail();

$mail->setBodyText($content, "UTF-8");

$mail->setBodyHtml($bodyHtml, "UTF-8");

$mail->setFrom('noreply@test.com', 'Admin');

//$mail->addTo("dao.th@sutrixmedia.com", 'Admin');

$mail->addTo("tran.hieu.dao@gmail.com", 'Admin');

$mail->setSubject("Test mail");

$result = $mail->send();

exit("ok");

}

}

**Embed image in email**

//$mail = new Zend\_Mail('utf-8');

/\*

$mail->setType(Zend\_Mime::MULTIPART\_RELATED);

$matches = array();

preg\_match\_all(

"#<img.\*?src=['\"]([^'\"]+)#i",

$bodyText,

$matches

);

$matches = array\_unique($matches[1]);

if (count($matches) > 0) {

foreach ($matches as $filename) {

$imagePath = $dir."/".$filename;

if (is\_readable($imagePath)) {

$at=$mail->createAttachment(file\_get\_contents($imagePath));

$extension = pathinfo($imagePath, PATHINFO\_EXTENSION);

switch ($extension) {

case 'gif':

$type = 'image/gif';

break;

case 'jpg':

case 'jpeg':

$type = 'image/jpg';

break;

case 'png':

$type = 'image/png';

break;

default:

$type = 'application/octet-stream';

}

$at->type = $type;

$at->disposition = Zend\_Mime::DISPOSITION\_INLINE;

$at->encoding = Zend\_Mime::ENCODING\_BASE64;

$at->id = 'cid\_' . md5\_file($imagePath);

$bodyText = str\_replace($filename, 'cid:' . $at->id, $bodyText);

}

}

}

$config = Zend\_Registry::get("configuration");

$mailto = $config->site->mailform->mailto;

$bodyText = str\_replace("[mailto]", $mailto,$bodyText);

$mail->setSubject('Login Information');

$mail->setFrom('noreply@test.com', 'Admin');

$mail->addTo($emailTo, 'Admin');

$mail->setBodyText($bodyCleanText);

$mail->setBodyHtml(

$bodyText,

'UTF-8',

'8bit'

);

//$mail->setBodyHtml($bodyText, null, '8bit');

\*

\*/