Namespace QHO\Library\Plugin;

Use Zend\Mvc\Controller\Plugin\AbstractPlugin;

Use Zend\Permissions\Acl\Acl;

Use Zend\Permissions\Acl\Role\GenericRole as Role;

Class AclPlugin extends AbstractPlugin{

Protected $role;

Public function getAuthService(){

$sm = $this->getController()->getServiceLocator();

$authService = $sm->get(‘authService’);

If($authService->hasIdentity()){

$user = $authService->getStorage()->read();

If($user[‘level’] == 1){

$this->role = ‘member’

}else{

$this->role = ‘admin’

}

}else{

$this->role = ‘guest’;

}

Return $this->role;

Public function RoleAccess($e){

$acl = new Acl;

$acl->deny();

$acl->addRole(new Role(‘guest’));

$acl->addRole(new Role(‘member’));

$acl->addRole(new Role(‘admin’));

$acl->addResource(‘training’)

->addResource(‘training:book’, ‘training’)

…

->addResource(‘training:user, ‘training’)

->addResource(‘training:verify, ‘training’);

$acl->allow(‘guest’ , ‘training:verify’ , array(‘index’ , ‘login’, ‘forgot’, ‘active’, ‘denied’));

$acl->allow(‘member’, ‘training:verify’ , array(‘logout’));

$acl->allow(‘member’ , ‘traiing:book’); …

$acl->allow(‘admin’);

$route = $e->getRouteMatch();

$controller = $route->getParam(‘controller’);

$moduleName = strtolower(substr($controller, 0 , strpos($controller, ‘\\’)));

$arr = explode(‘\\’ , $controller);

$controllerName = strtolower(array\_pop($arr));

$action = $route->getParam($action);

$role = $this->getAuthService();

If(!$acl->isAllowed($role, $moduleName . “:” .$controllerName, $actionName)){

$response = $e->getResponse();

$response->setStatusCode(302)->setContent(‘Access Denied’);

$response->sendHeaders();

If(isset($\_SERVER[‘HTTP\_X\_REQUESTED\_WITH’])){

$e->stopPropagation();

Module.php -> Onbootstrap();

$shared = $eventManager->getSharedManager();

$shared->attach(\_\_NAMESPACE\_\_ , ‘dispatch’, function($e){

$controller = $e->getTarget();

$route = $e->getRouteMatch();

$action = $route->getParam(‘action’);

If($controller instanceof Controller\VerifyController && $action != ‘logout’){

…

}else{

…

$sm = $this->getApplication()->getServiceManager;

$sm->get(‘ControllerPluginManager’)->get(‘QHO\Library\Plugin\AclPlugin’)->RoleAccess($e);

$response = $e->getResponse();

If($response->getStatusCode() == 302){

$e->stopPropagation();

$controller->plugin(‘redirect’)->toRoute(‘training/verify’,array(‘action’=> ‘denied’))