**Basic Auth Controller Class**

package com.maan.crm.auth.controller;

import org.apache.logging.log4j.LogManager;

import org.apache.logging.log4j.Logger;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import io.swagger.annotations.Api;

@RestController

@Api( tags="Basic Authentication", description = "API's.")

@RequestMapping("/basicauth")

public class BasicAuthController {

private Logger log = LogManager.getLogger(LoginController.class);

}

**Login Controller**

package com.maan.crm.auth.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.util.CollectionUtils;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import com.maan.crm.auth.dto.ChangePasswordReq;

import com.maan.crm.auth.dto.CommonLoginResponse;

import com.maan.crm.auth.dto.LoginRequest;

import com.maan.crm.auth.dto.LogoutRequest;

import com.maan.crm.auth.service.AuthendicationService;

import com.maan.crm.auth.service.LoginValidatedService;

import com.maan.crm.res.CommonCrmRes;

import com.maan.crm.util.error.CommonValidationException;

import com.maan.crm.util.error.Error;

import io.swagger.annotations.Api;

import io.swagger.annotations.ApiOperation;

@RestController

@Api( tags="Login Token Creation", description = "API.")

@RequestMapping("/authentication")

public class LoginController {

@Autowired

private AuthendicationService authservice;

@Autowired

private LoginValidatedService loginValidationComponent;

@PostMapping("/login")

@ApiOperation(value="This method is to Create Token For Access Other Apis")

public CommonLoginResponse getloginToken(@RequestBody LoginRequest mslogin, HttpServletRequest http) {

CommonLoginResponse res = new CommonLoginResponse();

List<Error> validation =loginValidationComponent.loginInputValidation(mslogin);

if(!CollectionUtils.isEmpty(validation)) {

res.setErrors(validation);

return res;

}

return authservice.checkUserLogin(mslogin,http);

}

@PostMapping("/changepassword")

@ApiOperation(value="This method is to Change Login Password")

public CommonCrmRes getChangePwd(@RequestBody ChangePasswordReq req) throws Exception {

CommonCrmRes res = new CommonCrmRes();

List<Error> error = loginValidationComponent.LoginChangePwdValidation(req);

if (error != null && error.size() > 0) {

throw new CommonValidationException(error, null);

}

res = authservice.LoginChangePassword(req);

return res;

}

@PostMapping("/logout")

@ApiOperation(value="This method is used to Logout From Screen")

public CommonCrmRes logout(@RequestBody LogoutRequest mslogin) {

return authservice.logout(mslogin);

}

}