```
using Meditech.App.Crm.Services;
using Meditech.App.Crm.ViewModels;
using Meditech.Core.Auth.Authorization;
using Meditech.Core.Auth.Entities;
using Meditech.Lib.Share;
using Meditech.Lib.Share.CommonClass.Paging;
using Microsoft.AspNetCore.Authorization;
using Microsoft.AspNetCore.Mvc;

namespace Meditech.App.Crm.Controllers
{
    [ApiController]
    [Route("{culture:culture}/api/v{version:apiVersion}/[controller]")]
    [ApiVersion("1.0")]
    public class ReportController : ControllerBase
    {
        private readonly ILogger<ReportController> _logger;
        private readonly IReportService _reportService;
        public ReportController(ILogger<ReportController> logger,
                                IReportService reportService)
        {
            _logger = logger;
            _reportService = reportService;
        }

        [HttpPost]
        [Route("chart-all-hospital")]
        [MediAuthorize(new[] { Lib.Share.PermitValue.ChoPhep }, Lib.Share.ActionPermit.Crm_Report_Kcb)]
        public IActionResult GetReportByAllBranch(ReportViewModel model)
        {
            try
            {
                var data = _reportService.GetReportByAllHospital(model);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"getReportByAllBranch error.");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }

        [HttpPost]
        [Route("chart-hospital")]
        [MediAuthorize(new[] { Lib.Share.PermitValue.ChoPhep }, Lib.Share.ActionPermit.Crm_Report_Kcb)]
        public IActionResult GetReportByBranch(ReportThisHospitalViewModel model)
        {
            try
            {
                var data = _reportService.GetReportByHospital(model);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"getReportByAllBranch error.");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }

        [HttpPost]
        [Route("feedback-from-customer")]
        public IActionResult GetFeedbackFromCustomer(ReportFeedbackFromCustomerViewModel model)
        {
            try
            {
                var data = _reportService.GetFeedbackFromCustomer(model);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"getReportByAllBranch error.");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }

        [HttpPost]
        [Route("feedback-from-customer/export")]
        public async Task<IActionResult> GetFeedbackFromCustomerExport(ReportFeedbackFromCustomerViewModel model)
        {
            try
            {
                var data = await _reportService.GetFeedbackFromCustomerExport(model);
                if (data.IsOk) return Ok(new { Data = data });
                return StatusCode(600, data);
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"GetFeedbackFromCustomerExport error.");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }

        [HttpPost]
        [Route("chart-kcb-all-hospital")]
        [MediAuthorize(new[] { Lib.Share.PermitValue.ChoPhep }, Lib.Share.ActionPermit.Crm_Report_Kcb)]
        public async Task<IActionResult> GetHoatDongKCBByAllHospital(ReportViewModel model)
        {
            try
            {
                var data = await _reportService.GetHoatDongKCBByAllHospital(model);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"getReportByAllBranch error.");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }

        [HttpPost]
        [Route("chart-kcb-hospital")]
        [MediAuthorize(new[] { Lib.Share.PermitValue.ChoPhep }, Lib.Share.ActionPermit.Crm_Report_Kcb)]
        public async Task<IActionResult> GetHoatDongKCBByHospital(ReportThisHospitalViewModel model)
        {
            try
            {
                var data = await _reportService.GetHoatDongKCBByHospital(model);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"getReportByBranch error.");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
    }
}
```