```
using Meditech.App.Crm.Services;
using Meditech.App.Crm.ViewModels;
using Meditech.Core.Auth.Authorization;
using Meditech.Lib.Share.CommonClass.HisGateway;
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 EsignController : ControllerBase
    {
        private readonly ILogger<EsignController> _logger;
        private readonly IMobileService _mobileService;
        private readonly IESignService _eSignService;
        public EsignController(ILogger<EsignController> logger,
                                IMobileService mobileService,
                                IESignService eSignService)
        {
            _logger = logger;
            _mobileService = mobileService;
            _eSignService = eSignService;
        }
        [MediAllowAnonymous]
        [AllowAnonymous]
        [HttpPost]
        [Route("check-the")]
        public async Task<IActionResult> CheckTheAsync([FromBody] CheckTheViewModel model)
        {
            try
            {
                //var requestUser = (BaseUser)Request.HttpContext.Items["User"];
                //if (requestUser == null) return Forbid();
                var data = await _mobileService.CheckThe(model, Guid.Empty);
                if (data.IsOk == false)
                {
                    return StatusCode(600, data);
                }
                return Ok(new { Data = data.Data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"CheckTheAsync error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
        [HttpPost]
        [Route("submit")]
        public IActionResult Submit([FromBody] ESignSubmitViewModel model)
        {
            try
            {
                var data = _eSignService.Submit(model);
                if (data.IsOk == false)
                {
                    return StatusCode(600, data);
                }
                return Ok();
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"Submit error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
        [HttpPost]
        [Route("find-esign")]
        public IActionResult FindEsign([FromBody] FindEsignViewModel model)
        {
            try
            {
                var data = _eSignService.FindEsign(model);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"Submit error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
    }
}
```