```
using Meditech.App.CommonService.Services;
using Meditech.App.CommonService.ViewModels;
using Meditech.Core.Auth.Authorization;
using Meditech.Core.Auth.Entities;
using Meditech.Lib.MediaStorage;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;

namespace Meditech.App.CommonService.Controllers
{
    [MediAuthorize]
    [Route("api/[controller]")]
    [ApiController]
    public class AttachmentController : ControllerBase
    {
        private readonly ILogger<AttachmentController> _logger;
        private readonly IMediaService _mediaService;
        private readonly IAttachmentService _attachmentService;
        public AttachmentController(ILogger<AttachmentController> logger,
                                IAttachmentService attachmentService,
                                IMediaService mediaService)
        {
            _logger = logger;
            _mediaService = mediaService;
            _attachmentService = attachmentService;
        }

        [HttpPost]
        [Route("upload")]
        public async Task<IActionResult> UploadFile([FromForm] MediaViewModel model)
        {
            try
            {
                var data = await _mediaService.UploadFileV2(model);
                if (data.IsOk == false) return StatusCode(600, data);
                return Ok(new { Data = data.Data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"UploadFile error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
        [HttpGet]
        [Route("get-by-object-id")]
        public IActionResult GetByObjectId([FromQuery]string objectId, [FromQuery] string objectType, [FromQuery] string? searchText)
        {
            try
            {
                var data = _attachmentService.GetByObjectId(objectId, objectType, searchText);
                return Ok(new { Data = data });
            }
            catch(Exception ex)
            {
                _logger.LogError(ex, $"GetByObjectId error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
        [HttpPost]
        public IActionResult Add([FromBody] List<AttachmentAddViewModel> model)
        {
            try
            {
                var requestUser = (BaseUser)Request.HttpContext.Items["User"];
                if (requestUser == null)
                    return Forbid();
                var ret = _attachmentService.Add(model, requestUser.Id);
                if (ret.IsOk == false) return StatusCode(600, ret);
                return Ok();
            }
            catch(Exception ex)
            {
                _logger.LogError(ex, $"GetByObjectId error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
        [HttpPut]
        public IActionResult Update([FromBody] AttachmentUpdateViewModel model)
        {
            try
            {
                var requestUser = (BaseUser)Request.HttpContext.Items["User"];
                if (requestUser == null)
                    return Forbid();
                var ret = _attachmentService.Update(model, requestUser.Id);
                if (ret.IsOk == false) return StatusCode(600, ret);
                return Ok();
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"Update error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
        [HttpDelete("id")]
        public IActionResult Delete(Guid id)
        {
            try
            {
                var requestUser = (BaseUser)Request.HttpContext.Items["User"];
                if (requestUser == null)
                    return Forbid();
                var ret = _attachmentService.Delete(id, requestUser.Id);
                if (ret.IsOk == false) return StatusCode(600, ret);
                return Ok();
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"Delete error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
    }
}
```