

```

using Meditech.App.Crm.Services;
using Meditech.App.Crm.ViewModels;
using Meditech.Core.Auth.Authorization;
using Meditech.Core.Auth.Entities;
using Microsoft.AspNetCore.Mvc;

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

        [HttpPost]
        [Route("search-grid")]
        public IActionResult SearchGrid(SearchGridDoctorViewModel model)
        {
            try
            {
                var requestUser = (BaseUser)Request.HttpContext.Items["User"];
                if (requestUser == null) return Forbid();
                var data = _doctorManagementService.SearchGrid(model);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"SearchGrid error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }

        [HttpGet]
        [Route("mobile/get-all")]
        public IActionResult GetAlldoctor()
        {
            try
            {
                var requestUser = (BaseUser)Request.HttpContext.Items["User"];
                if (requestUser == null) return Forbid();
                var data = _doctorManagementService.GetAllDoctor();
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"GetAlldoctor error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }

        [HttpGet]
        [Route("get-by-id")]
        public IActionResult GetdoctorById(Guid id)
        {
            try
            {
                var requestUser = (BaseUser)Request.HttpContext.Items["User"];
                if (requestUser == null) return Forbid();
                var data = _doctorManagementService.getDoctorById(id);
                return Ok(new { Data = data });
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"GetdoctorById error. ");
                return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
            }
        }
    }
}

```

```

    {
        _logger.LogError(ex, $"GetdoctorById error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

[HttpPost]
public IActionResult AddNewdoctorManagement(DoctorManagementViewModel model)
{
    try
    {
        var requestUser = (BaseUser)Request.HttpContext.Items["User"];
        if (requestUser == null) return Forbid();
        var data = _doctorManagementService.AddNewDoctorManagement(model, requestUser.Id);
        return Ok(new { Data = data });
    }
    catch (Exception ex)
    {
        _logger.LogError(ex, $"AddNewdoctorManagement error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

[HttpPut]
public IActionResult UpdatedoctorManagement(UpdateDoctorManagementViewModel model)
{
    try
    {
        var requestUser = (BaseUser)Request.HttpContext.Items["User"];
        if (requestUser == null) return Forbid();
        var data = _doctorManagementService.UpdateDoctorManagement(model);
        return Ok(new { Data = data });
    }
    catch (Exception ex)
    {
        _logger.LogError(ex, $"UpdatedoctorManagement error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

[HttpDelete]
public IActionResult DeletedoctorManagement(Guid id)
{
    try
    {
        var requestUser = (BaseUser)Request.HttpContext.Items["User"];
        if (requestUser == null) return Forbid();
        var data = _doctorManagementService.DeleteDoctorManagement(id);
        return Ok(new { Data = data });
    }
    catch (Exception ex)
    {
        _logger.LogError(ex, $"DeletedoctorManagement error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

[HttpPut]
[Route("change-state")]
public IActionResult StatusDoctorManagement(UpdateDoctorManagementViewModel model)
{
    try
    {
        var requestUser = (BaseUser)Request.HttpContext.Items["User"];
        if (requestUser == null) return Forbid();
        var data = _doctorManagementService.ChangStateDoctorManagement(model);
        return Ok(new { Data = data });
    }
    catch (Exception ex)
    {
        _logger.LogError(ex, $"StatusDoctorManagement error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

```

```

[HttpGet]
[Route("get-comment-by-id")]
public IActionResult GetDoctorCommentById(Guid id)
{
    try
    {
        var requestUser = (BaseUser)Request.HttpContext.Items["User"];
        if (requestUser == null) return Forbid();
        var data = _doctorManagementService.GetCommentList(id);
        return Ok(new { Data = data });
    }
    catch (Exception ex)
    {
        _logger.LogError(ex, $"GetDoctorCommentById error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

```

```

[HttpDelete]
[Route("delete-comment-by-id")]
public IActionResult DeleteDoctorCommentById(Guid id)
{
    try
    {
        var requestUser = (BaseUser)Request.HttpContext.Items["User"];
        if (requestUser == null) return Forbid();
        var data = _doctorManagementService.DeleteCommentById(id);
        return Ok(new { Data = data });
    }
    catch (Exception ex)
    {
        _logger.LogError(ex, $"DeleteDoctorCommentById error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

```

```

[HttpPost]
[Route("print-qr-code")]
public IActionResult GetDoctorPrintQrCode(List<Guid> ids)
{
    try
    {
        var requestUser = (BaseUser)Request.HttpContext.Items["User"];
        if (requestUser == null) return Forbid();
        var data = _doctorManagementService.GetDoctorPrintQrCode(ids);
        return Ok(new { Data = data });
    }
    catch (Exception ex)
    {
        _logger.LogError(ex, $"GetDoctorPrintQrCode error. ");
        return StatusCode(600, $"Lỗi hệ thống. Error={ex.Message}");
    }
}

```

```

}
}

```