

```

using Medical.DTO.Cms1500;
using Medical.Model;
using Microsoft.EntityFrameworkCore;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace Medical.BLL
{
    public class MedicalService : ServiceManager<Cms1500FormDTO, Cms1500SearchDTO>
    {
        private readonly MedicalContext? _context;

        public MedicalService() { }

        public MedicalService(MedicalContext context)
        {
            _context = context;
        }

        public override async Task DeleteItemAsync(Cms1500FormDTO dto)
        {
            Response.ErrorList.Clear();
            try
            {
                if (_context == null) return;

                var form = await _context.CMS1500Forms
                    .FirstOrDefaultAsync(f => f.FormId == dto.FormId);

                if (form != null)
                {
                    _context.CMS1500Forms.Remove(form);
                    await _context.SaveChangesAsync();
                }
            }
            catch (Exception ex)
            {
                Response.ErrorList.Add(new Error { Code = ex.Source, Message = ex.Message });
            }
        }

        public override async Task GetItemAsync(Cms1500SearchDTO searchParams)

```

```

{
    Response.ErrorList.Clear();
    try
    {
        if (_context == null) return;

        var form = await _context.CMS1500Forms
            .Include(f => f.DiagnosisCodes)
            .Include(f => f.ServiceLines)
            .FirstOrDefaultAsync(f => f.FormId == searchParams.FormId);

        Response.Item = MedicalDTOParser.ParseDTO(form);
    }
    catch (Exception ex)
    {
        Response.ErrorList.Add(new Error { Code = ex.Source, Message = ex.Message });
    }
}

public override async Task GetListAsync(Cms1500SearchDTO searchParams)
{
    try
    {
        if (_context == null) return;

        var query = _context.CMS1500Forms.AsQueryable();

        if (searchParams.FormId.HasValue)
            query = query.Where(f => f.FormId == searchParams.FormId);

        if (searchParams.CreatedDate.HasValue)
            query = query.Where(f => f.CreatedDate >= searchParams.CreatedDate);

        var totalItems = await query.CountAsync();
        var forms = await query.Skip(searchParams.PageNumber * searchParams.PageSize)
            .Take(searchParams.PageSize)
            .ToListAsync();

        Response.List = forms.Select(MedicalDTOParser.ParseDTO).ToList();
        Response.TotalItems = totalItems;
    }
    catch (Exception ex)
    {
        Response.ErrorList.Add(new Error { Code = ex.Source, Message = ex.Message });
    }
}

```

```

    }
}

public override async Task SaveItemAsync(Cms1500FormDTO dto)
{
    Response.ErrorList.Clear();
    try
    {
        if (_context == null) return;

        var form = await _context.CMS1500Forms
            .Include(f => f.DiagnosisCodes)
            .Include(f => f.ServiceLines)
            .FirstOrDefaultAsync(f => f.FormId == dto.FormId);

        if (form == null)
        {
            form = new CMS1500Form();
            MedicalModelParser.ParseModel(form, dto);
            form.CreatedDate = DateTime.UtcNow;
            _context.CMS1500Forms.Add(form);
        }
        else
        {
            MedicalModelParser.ParseModel(form, dto);
            form.UpdatedDate = DateTime.UtcNow;
        }

        await _context.SaveChangesAsync();
    }
    catch (Exception ex)
    {
        Response.ErrorList.Add(new Error { Code = ex.Source, Message = ex.Message });
    }
}
}

```

```

public class Cms1500SearchDTO : SearchDTO
{
    public int? FormId { get; set; }
    public DateTime? CreatedDate { get; set; }
    public DateTime? ServiceDateFrom { get; set; }
    public DateTime? ServiceDateTo { get; set; }
}

```

}