```csharp
using Meditech.Core.Infrastructure.Database;
using Meditech.Lib.CommonShare;
using Meditech.Lib.CommonShare.Extensions;
using Meditech.Lib.Firebase;
using Microsoft.Extensions.Configuration;
using Newtonsoft.Json;

namespace Meditech.App.CommonService.Services
{
    public interface IDataInitalizeService
    {
        void SystemInitialize();
    }
    public class DataInitalizeService : IDataInitalizeService
    {
        private readonly ILogger<DataInitalizeService> _logger;
        private readonly IConfiguration _configuration;
        private readonly ISignatureProcessService _signatureProcessService;
        public DataInitalizeService(ILogger<DataInitalizeService> logger,
                                    ISignatureProcessService signatureProcessService,
                                    IConfiguration configuration)
        {

            _logger = logger;
            _configuration = configuration;
            _signatureProcessService = signatureProcessService;
        }

        public void SystemInitialize()
        {
            InitFirebaseMessage();
            try
            {
                _signatureProcessService.InitDocumentTypes(Guid.Empty);
                _logger.LogInformation("InitDocumentTypes done");
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"InitDocumentTypes error");
            }
            try
            {
                _signatureProcessService.InitSignPositions();
                _logger.LogInformation("InitSignPositions done");
            }
            catch (Exception ex)
            {
                _logger.LogError(ex, $"InitSignPositions error");
            }
        }

        private void InitFirebaseMessage()
        {
            try
            {
                IMessageService messageService = FirebaseMessageService.GetInstance();
                Dictionary<string, object> settings = _configuration
                    .GetSection("firebaseAdminKey")
                    .Get<Dictionary<string, object>>();
                string credential = JsonConvert.SerializeObject(settings);
                // string credential = _configuration.GetSection("firebaseAdminKey").Value.ToString();
                bool setUpDone = messageService.SetupEnv(credential);
                if (setUpDone)
                {
                    _logger.LogInformation($"Start firebase service success");
                }
                else
                {
                    _logger.LogWarning($"Start firebase service failed");
                }
```

```
            }
            catch(Exception ex)
            {
                _logger.LogError(ex, "InitFirebaseMessage error");
            }
        }
    }
}
```