Fund Administration API

Tools and Tech

- Architecture Pattern - Clean Architecture
- Design Principles - Domain driven design, Solid, Repository pattern, CQRS
- External NuGet packages -
  - MediatR
  - FluentValidation
  - Ardalis packages
  - Serilog packages
  - Nsubstutite

Installation Guide

1. Update SqlConnectionString in appsettings.json
2. Build complete solution
3. Set FundAdministration.API project as a startup project
4. Run (F5) the solution
5. Swagger UI should be loaded
6. Check if the db is created with tables in the sql server and seed data (the db gets created automatically on startup due to this statement context.Database.EnsureCreated())

Features Covered

1. **Tests**
   a) The following tests covered for Fund Feature
      i. *Entity Class* - FundTests
      ii. *Service Class(Use Case)* **-**
         1. CreateFundHandlerTest
         2. CreateFundValidatorTest
         3. DeleteFundHandlerTest
         4. GetFundHandlerTest
         5. ListFundHandlerTest
         6. UpdateFundHandlerTest
      iii. *Controller Class -* FundControllerTest
      iv. *Repository class -* FundRepositoryTests (using In memory db)
2. **Assignment feature -**
   a) Implemented all the functional features mentioned in the assignment (including optional ones)
3. **Documentation** -
   a) Created Uml class diagram - for showing the classes and entity invloved for fund functionalies(CURD operations)
   b) Created uml sequence diagram for showing the general data flow in the application with mediatR.
   c) Added Readme documentation(this document)
4. **Room for Improvement** (did not do due to time constraints)
   a) Integration testing for API endpoints using real db - I will use **TestContainers** NuGet package which can be used with docker which generates db per tests and destory it once the test is done.
   b) Swagger documentation for DTOs
   c) Integration testing for middleware which captures and log error.