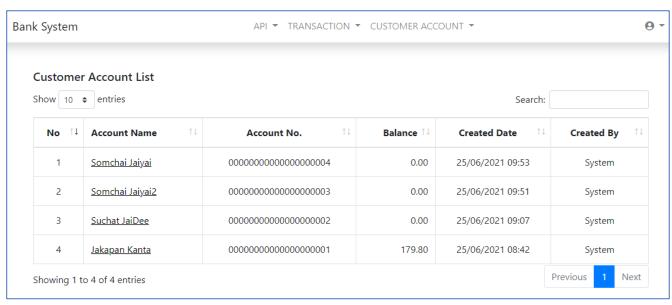
Bank Web API

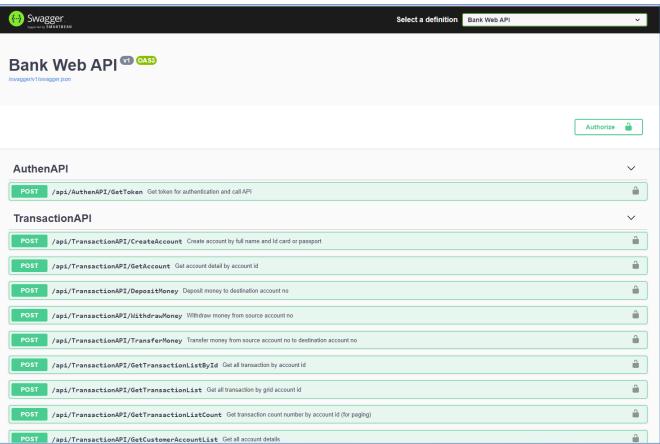
Example of ASP.Net Core 3.1, MVC, C#, API, JWT, EF Core, LINQ, jQuery, Bootstrap, MSTest, Selenium

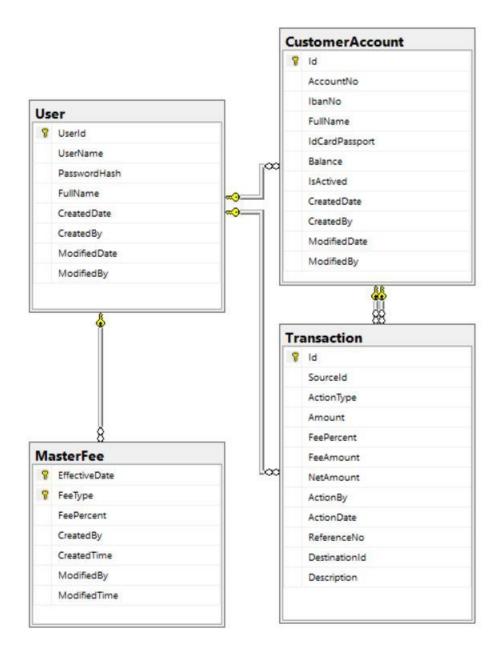
Database: SQL Server Express LocalDB, Database Design, PK, FK, Index, Paging

Function: Create account, Show details, Deposit, Withdraw, Transfer, Fee, Transaction List

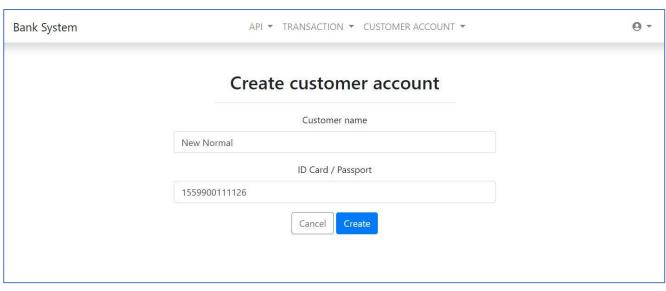
Source code: https://github.com/Ligerking007/BankWebAPI

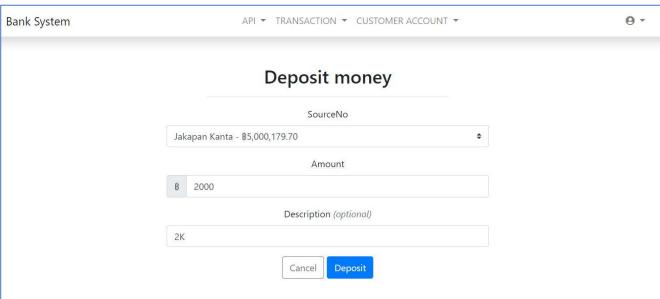


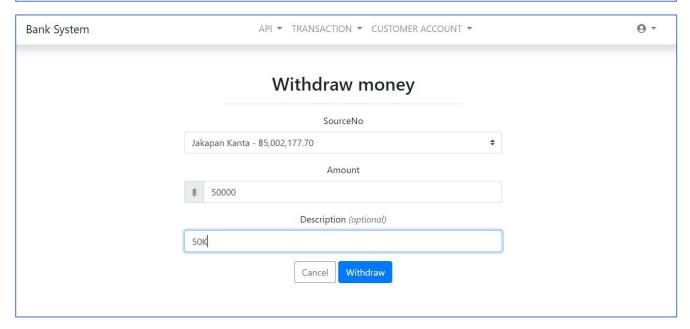


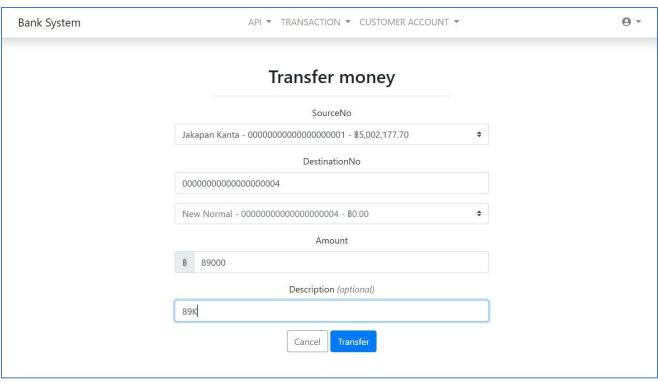


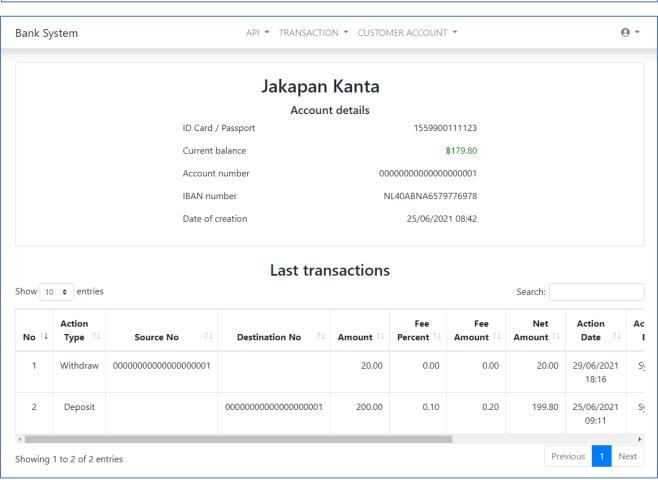
Database Design, PK, FK, Index











Source Code Example

```
| December | Proceedings | Improvement | Proceding | P
```

Authentication API

```
| TableWindows | Table Windows | Table Windows
```

Controller API

```
merAccountService.cs + X CustomerAccountTest.cs
                                                                       Transfer.cshtml
                                                                                                 _ViewStart.cshtml _Layout.cshtml
                                                                                                                                                                C# Core.Services
                                                                                                                                                                         private ITransactionRepository _ITransactionRepository;
private IMasterFeeRepository _IMasterFeeRepository;
SQL
                                                                                                                                                                          Search Solution Explorer (Ctrl+;)
          26
27
                                                                                                                                                                          Solution 'BankWebAPI' (9 of 9 projects)
                                private IMapper _IMapper;
                                                                                                                                                                             Core

Core.Common
                               public CustomerAccountService(ILogger<CustomerAccountService> _ILogger,
                                    ICustomerAccountRepository _ICustomerAccountRepository,
IFransactionRepository _ITransactionRepository,
IMasterFeeRepository _IMasterFeeRepository,
IMapper _IMapper)
                                                                                                                                                                             D a C# Core.Interfaces
D a C# Core.Models
                                                                                                                                                                              ▲ a C# Core.Services
                                                                                                                                                                                Dependencies
Authen
Bank
                                    this._ILogger = _ILogger;
this._ICustomerAccountRepository = _ICustomerAccountRepository;
this._ITnansactionRepository = _ITransactionRepository;
this._IMasterFeeRepository = _IMasterFeeRepository;
                                                                                                                                                                                 Repository
                                     this. IMapper = IMapper;
                                                                                                                                                                             ▶ a C# Repository.Entities
                                                                                                                                                                              ▶ a C# Repository.Interfaces
                                                                                                                                                                              ▶ a C# Repository.Models

■ UnitTest

                                public BaseResponse CreateCustomerAccount(CustomerAccountModel req)

■ ■ UnitTest.Services

                                                                                                                                                                                 Dependencies
                                     BaseResponse res = new BaseResponse() { IsSuccess = false };
                                                                                                                                                                                 D & C# CommonTest.cs
           45
                                                                                                                                                                                  a C# CustomerAccountTest.cs
           47
48
49
                                         var modelUpdate = _ICustomerAccountRepository.AsQueryable()
    .FirstorDefault(f => f.Id == req.Id ||
    (f.FullName== req.FullName && f.IdCardPassport==req.IdCardPassport));//Check Duplicate
if (modelUpdate != null)

▲ BankWebAPI

                                                                                                                                                                              Dependencies
                                                                                                                                                                              ▶ a 🖣 Properties
                                                                                                                                                                              ⊿ a ⊕ www.root
                                              modelUpdate.FullName = req.FullName;
modelUpdate.IdCardPassport = req.IdCardPassport;
modelUpdate.ModifiedBy = req.CreatedBy;
modelUpdate.ModifiedDate = DateTime.Now;
                                                                                                                                                                                 Þ a ima
                                                                                                                                                                                   a 🔝 favicon.ico

▲ a □ Controllers

                                               _ICustomerAccountRepository.Update(modelUpdate, true);
                                                                                                                                                                                🛮 a 🗀 API
                                                                                                                                                                                    ▶ a C* TransactionAPIController.cs

    b a C* BaseController.cs
    b a C* CustomerAccountController.cs

                                               req.AccountNo = GenerateAccountNo();
                                               req.IbanNo = GenerateIBANNo();
                                                                                                                                                                                 ▶ a C* HomeController.cs
                                              req.CreatedDate = DateTime.Now;
req.CreatedBy = req.CreatedBy;
req.CreatedBy = req.CreatedBy;
vam modelDB = IMapper.MapcCustomerAccount>(req.ICustomerAccountRepository.Add(modelDB, true);
                                                                                                                                                                              ▶ a i Models

■ a □ Views

                                                                                                                                                                                 ▶ a ☐ CustomerAccount
           68
69
                                                                                                                                                                                 ▶ a i Errors
                                                                                                                                                                                  ▶ a iii Home
                                         res.IsSuccess = true;
res.Message = "Completed";

ViewStart.cshtml

ViewStart.cshtml
                                     catch (Exception ex)
                                         res.IsSuccess = false;
res.Message = ex.Message;
_ILogger.Error(ex);
```

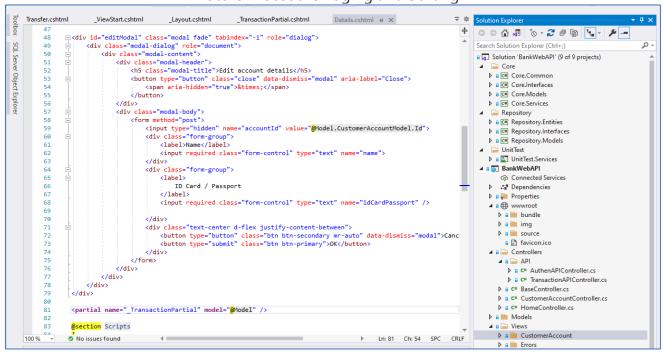
Dependency Injection & Services

```
| Second continue | Second con
```

Server Paging and Sorting

```
--Execute [dbo].[SP_GetCustomerAccountList] '1','Id','Desc','0','2' ,1
ALTER PROCEDURE [dbo].[SP_GetCustomerAccountList]
     @IsActived bit = 1
     @SortBy nvarchar (100) = null ,
     @SortDirection nvarchar (10) = null ,
     @PageStart int = 0 ,
     @PageSize int = 1000 ,
     @TotalRows int output
 ÁS
BEGIN
       - Build WHERE Clause
     DECLARE @WhereClause [nvarchar] (2000)
     SET @WhereClause = ' IsActived = ''' + convert(nvarchar, @IsActived) + ''' '
     DECLARE @OrderBy [nvarchar] (100)
SET @OrderBy = @SortBy + ' ' + @SortDirectIF (@SortBy is null or LEN(@SortBy) = 0)
                                       + @SortDirection
     BEGIN
     -- default order by to first column
     SET @OrderBy = 'Id Desc'
     END
       - SQL Server 2005 Paging
     declare @SqlCount as nvarchar(max)
     declare @SqlTable as nvarchar(max)
     declare @SQL as nvarchar(max)
     -- get row count
SET @SqlCount = ' SELECT @TotalRows = COUNT(*)'
     -- sql : from table and where ...
SET @SqlTable = ' FROM [BankDB].[dbo].[CustomerAccount] '
     IF LEN(@WhereClause) > 0
     BEGIN
          SET @SqlTable = @SqlTable + ' WHERE ' + @WhereClause
     END
     SET @SqlCount = @SqlCount + @SqlTable
           EXEC sp_executeq1 @SqlCount , N'@TotalRows int OUTPUT', @TotalRows OUTPUT ; --print @TotalRows;
          -- Get Data
SET @SQL = ' SELECT'
          SET @SQL = @SQL + ' ROW_NUMBER() OVER (ORDER BY ' + @OrderBy + ') as RowIndex' SET @SQL = @SQL + ' , *^*--All field
           -- query @SqlTable : from table where
          SET @SQL = @SQL + @SqlTable
SET @SQL = @SQL + ' ORDER BY ' + @OrderBy
          SET @SQL = @SQL + ' OFFSET '+ convert(nvarchar, @PageStart)+' ROWS FETCH NEXT '+convert(nvarchar, @PageSize)+' ROWS ONLY'
          -- Get Data
          EXEC sp_executesql @SQL
--PRINT @SQL
END
```

Store Procedure Paging and Sorting



HTML & Bootstrap

Jquery

```
TransactionRepository.cs CustomerAccountService.cs
                          × UserRepository.cs
   C# Repository.Entities
                                                  🔻 🔩 Repository.Entities.GenericRepository<TContex 🕶 🔩 Entities
                                                                                                                                                   + +
                                                                                                                                                         G O A A TO - 2 A B TU - 1
SQL
                                                                                                                                                         Search Solution Explorer (Ctrl+;)
                                                                                                                                                         a . Solution 'BankWebAPI' (9 of 9 projects)
                                                                                                                                                         ▲ Core
Core.Common
         10
11
                       public abstract class GenericRepository<TContext, TEntity> : IGenericRepository<TEntity>
                            where TEntity : class
where TContext : DbContext//, new()
                                                                                                                                                             ▶ a C# Core.Interfaces
                                                                                                                                                             D a C■ Core.Models
                            // private readonly MCSurveyContext _dbContext;
protected TContext Entities;
protected DbSet<TEntity> DataSet;
                                                                                                                                                             ▲ a C# Core.Services
                                                                                                                                                               Dependencies

Authen
         16
                                                                                                                                                                  public GenericRepository(TContext context)...
                                                                                                                                                             ▲ a C# Repository.Entities
                            public virtual IQueryable<TEntity> AsQueryable()...
                                                                                                                                                               Dependencies

BankDB
                                                                                                                                                                  ▶ a C* CustomerAccountRepository.cs
         29
                            public virtual IQueryable<TEntity> FromSql(string sql)...
                                                                                                                                                                   public virtual IQueryable<TEntity> GetAll()...

    b a C<sup>®</sup> TransactionRepository.cs
    b a C<sup>®</sup> UserRepository.cs

                                                                                                                                                                  BankDB_Entity.tt
BankDB_Entity.txt
         38
42
                            public virtual IQueryable<TEntity> Database()...

    ▶ a C# GenericRepository.cs
    ▶ a C# Repository.Interfaces

                            public virtual DbContext Entity()...
         47
                                                                                                                                                             ▶ a C# Repository.Models
                            public virtual EntityEntry<TEntity> Add(TEntity entity, bool IsCommit = false)
                                                                                                                                                             ▲ a ☑ UnitTest,Services
         49
                                                                                                                                                                Dependencies
                                 EntityEntry<TEntity> newEntry = null;
                                rettity.ser.y...
try {
    newEntry = DataSet.Add(entity);
    Entities.Entry(entity).State = EntityState.Added;
    if(IsCommit){
        Entities.SaveChanges();
    }
}
                                                                                                                                                                C# CommonTest.cs
                                                                                                                                                          ▲ a ■ BankWebAPI
                                                                                                                                                                 (%) Connected Services
                                                                                                                                                             Dependencies
                                                                                                                                                             Properties

www.root
                                 }catch(Exception ex){
   throw ex;
                                                                                                                                                               ▷ a bundle
▷ a img
         59
60
                                 return newEntry;
                                                                                                                                                                  a 🔝 favicon.ico
         62
63
                                                                                                                                                             ■ Controllers
■ □ API
                            public virtual EntityEntry<TEntity> Update(TEntity entity, bool IsCommit = false)
                                                                                                                                                                  ▶ a C* AuthenAPIController.cs
         65
66
67
68
69
                                                                                                                                                                   EntityEntry<TEntity> updateEntry = null;
                                                                                                                                                                ▶ a C# BaseController.cs
                                                                                                                                                                updateEntry = DataSet.Update(entity);
                                                                                                                                                                ▶ a C# HomeController.cs
                                     Entities.Entry(entity).State = EntityState.Modified;
if (IsCommit)

■ G Wiews

    ▶ Ln: 1 Ch: 1 SPC CRLF

Ln: 1 Ch: 1 SPC CRLF

Solution Explorer Git Changes
```

Repository (Connect Database)

```
_Layout.cshtml
                                                                     _ViewStart.cshtml
                                                                                                                             _TransactionPartial.cshtml

  ▼
  Φ

  Solution Explorer

     ☑ UnitTest.Services
                                                                                                                                                                                           SQL Server Object Explore
                                                                                                                                                                                           Search Solution Explorer (Ctrl++)
            86
87
                                   [TestMethod]
                                                                                                                                                                                           a 🔊 Solution 'BankWebAPI' (9 of 9 projects)
                                   public void CalculateBalanceDeposit_0_1000()
                                                                                                                                                                                               ▶ a C# Core.Common
           89
90
91
92
93
94
95
96
97
98
99
                                                                                                                                                                                                ▶ a C# Core.Interfaces
                                        Initialize();
                                        ▶ a C# Core.Models
                                                                                                                                                                                                ▶ a C# Core.Services

▲ Repository

    D a C# Repository.Entities
    D a C# Repository.Interfaces

                                                                                                                                                                                                ▶ a C# Repository.Models
                                                                                                                                                                                               UnitTest

UnitTest.Services
                                                                                                                                                                                                    Dependencies
C CommonTest.cs
                                   [TestMethod]
           101
102
103
104
105
106
107
108
109
110
111
112
113
                                   public void CalculateBalanceDeposit_500_1000()
                                                                                                                                                                                                   ■ BankWebAPI
                                         Initialize();
                                        Initialize();

var __ICUSTOMER'ACCOUNTSERVICE = _kernel.Get<ICustomerAccountService>();

decimal balance = 500;

decimal amount = 1000;

decimal feePercent = 0.1m;

decimal feePercent = _ICustomerAccountService.CalculateFeeAmount(amount, feePercent); //1 //Calcul

decimal neAmount = _ICustomerAccountService.CalculateWetAmountDeposit(amount, feeAmount);//1999 /

decimal neW8alance = _ICustomerAccountService.CalculateBalanceDeposit(balance, netAmount);//1499

Assert.AreFacual(new8alance, 1499m);
                                                                                                                                                                                                     (A) Connected Services
                                                                                                                                                                                                Dependencies
                                                                                                                                                                                                ▶ a 🖣 Properties
                                                                                                                                                                                                ⊿ a ⊕ www
                                                                                                                                                                                                   bab img
a source
                                         Assert.AreEqual(newBalance, 1499m);
                                                                                                                                                                                                      a 🖟 favicon.ico

▲ a  Controllers

                                   #endregion
#region Generate
[TestMethod]
                                                                                                                                                                                                    🔺 🔒 🚞 API

    D a C* AuthenAPIController.cs
    D a C* TransactionAPIController.cs

           116
                                    public void GenerateIBANNo()

    b a C* BaseController.cs
    b a C* CustomerAccountController.cs

           117
                      ▶ a C# HomeController.cs
                                                                                                                                                                             . т Д 🗙

■ G □ Views

                                                                                        ▶ a ■ CustomerAccount
      Show output from: Package Manager
```

Unit Test

Other Technique

Memory controlling

Dispose object (clears memory)

Garbage collection (clears memory)

String builder

Date conversion

Null validation

SQL Transaction (Begin, Commit, Rollback)

Reduce looping

Reduce connect database

Security coding (OWASP10), JWT

OOP concept

Clean code

Load balancing

Scaling