|  |  |
| --- | --- |
| Bank | Bank details |
| BankAccount | Bank Account details |
| Transaction / ScheduledTransaction / Movement  Transfer | Defines a movement of money from one account to another. May be one-off or repeating (e.g. monthly).  Name  FromBankAccount, ToBankAccount,  Amount, AmountTolerence  StartDate, EndDate, Frequency |
| Cashflow | Defines the starting point for a cashflow projection.  Name, List of bankAccount, Date, Opening Balance |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Feature \ Module** | **Bank** | **BankAccount** | **Transfer** | **BankTree** | **Cashflow** |
| **VM Edit to wrap Entity** |  |  |  |  | Y |
| **VM Item to wrap Entity** |  |  |  |  |  |
| **Agent class to handle Add/Edit** |  |  |  |  | Y |
| **Delete multipe ids** |  |  |  |  |  |
| **EF6** |  |  |  |  |  |
| **EF6 extended for delete multiple** |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| **Make account name property same?** |  |  |  |  |  |
| **Tasks for background** |  |  |  |  |  |

**TO DO**

|  |  |
| --- | --- |
| Error message dialog is not modal | Y |
| Use Item property in editor instead of individual properties? |  |
| Do we still need IGenericRepository? |  |
| List of accounts for cashflow editor |  |
| Pressing Enter on OK in Add/Edit will re-open the dialog |  |
|  |  |

When to create an interface for a class?

Anything added to the DI container. E.g. Services, Factories

New entity process

|  |  |
| --- | --- |
| Create table | Y |
| Generate EF entity | Y |
| Core entity | Y |
| AutoMapper Mappings for EF->Core | Y |
| Unit test for mappings | Y |
| Repository | Y |
| Integration Test for repo | Y |
| Item ViewModels | Y |
| Mappings for Core -> Item VM | Y |
| Unit tests for Item VM mappings | Y |
| List ViewModel (without editor) | Y |
| Add to DI | Y |
| List View | Y |
| Editor ViewModel | Y |
| Mappings for Core -> Editor VM | Y |
| Unit tests for Editor VM mappings | Y |
| VM Editor factory interface | Y |
| Add editor to List VM | Y |
| Editor View |  |
|  |  |
|  |  |

Notes

Model.Context.tt

public <#=code.Escape(container)#>(string nameOrConnectionString)

: base(nameOrConnectionString)

{

<#

if (!loader.IsLazyLoadingEnabled(container))

{

#>

this.Configuration.LazyLoadingEnabled = false;

<#

}

#>

}