## «Interface»
### IGenericRepository<TEntity>

+Delete(Tentity)
+Delete(object)
+ Get(Expression<Func<TEntity, bool>> filter = null, Func<IQueryable<TEntity>, IOrderedQueryable<TEntity>> orderBy = null, string includeProperties = ""): IEnumerable<TEntity>
+GetByID(object): Tentity
+Insert(Tentity)
+Update(Tentity)

## «class»
### GenericRepository<TEntity>

+Delete(Tentity)
+Delete(object)
+ Get(Expression<Func<TEntity, bool>> filter = null, Func<IQueryable<TEntity>, IOrderedQueryable<TEntity>> orderBy = null, string includeProperties = ""): IEnumerable<TEntity>
+GetByID(object): Tentity
+Insert(Tentity)
+Update(Tentity)

```
                      «Interface»
                      IUnitOfWork

+Save()
+Dispose()
+GenericRepository<Address> { get; }
+IGenericRepository<BarterAdd> { get; }
+IGenericRepository<Comment> { get; }
+IGenericRepository<TradeHistory>{ get; }
+IGenericRepository<TradeRequest> { get; }
+IGenericRepository<ApplicationUser>{ get; }
```

```
                      «Interface»
                      IUnitOfWork

+Save()
+Dispose()
+GenericRepository<Address> { get; }
+IGenericRepository<BarterAdd> { get; }
+IGenericRepository<Comment> { get; }
+IGenericRepository<TradeHistory>{ get; }
+IGenericRepository<TradeRequest> { get; }
+IGenericRepository<ApplicationUser>{ get; }
-ApplicationDbContext context
-IGenericRepository<BarterAdd>
-IGenericRepository<Address>
-IGenericRepository<Comment>
-IGenericRepository<TradeRequest>
-IGenericRepository<ApplicationUser>
-IGenericRepository<TradeHistory>
-IGenericRepository<FinishedTrade>
-IGenericRepository<Rating>
```