

Namespace Raftel.Application.Abstractions

Classes

[RequestDispatcher](#)

Default implementation of [IRequestDispatcher](#) that resolves the request handler and middleware pipeline from the dependency injection container.

Interfaces

[IRequestDispatcher](#)

Dispatches requests through the pipeline and to their associated handler.

[IRequestHandler<TRequest, TResponse>](#)

Defines a handler that processes a request of type **TRequest** and returns a response of type **TResponse**.

[IRequestMiddleware<TRequest, TResponse>](#)

Represents a middleware component in the request pipeline that can intercept and modify the processing of a request before and/or after the next component is invoked.

[IRequest<TResponse>](#)

Represents a request that produces a response of type **TResponse**.

Delegates

[RequestHandlerDelegate<TResponse>](#)

Delegate that represents the next handler or middleware in the request processing pipeline.

Interface IRequestDispatcher

Namespace: [Raftel.Application.Abstractions](#)

Assembly: Raftel.Application.dll

Dispatches requests through the pipeline and to their associated handler.

```
public interface IRequestDispatcher
```

Methods

DispatchAsync<TRequest, TResponse>(TRequest)

Dispatches the given request and returns the result asynchronously.

```
Task<TResponse> DispatchAsync<TRequest, TResponse>(TRequest request) where TRequest  
: IRequest<TResponse>
```

Parameters

request TRequest

The request to dispatch.

Returns

[Task](#)  <TResponse>

The result returned from the request handler or middleware.

Type Parameters

TRequest

The type of the request.

TResponse

The type of the response.

Interface IRequestHandler<TRequest, TResponse>

Namespace: [Raftel.Application.Abstractions](#)

Assembly: Raftel.Application.dll

Defines a handler that processes a request of type **TRequest** and returns a response of type **TResponse**.

```
public interface IRequestHandler<TRequest, TResponse> where TRequest : IRequest<TResponse>
```

Type Parameters

TRequest

The type of the request.

TResponse

The type of the response.

Methods

HandleAsync(TRequest)

Handles the given request asynchronously.

```
Task<TResponse> HandleAsync(TRequest request)
```

Parameters

request TRequest

The request to process.

Returns

[Task](#) <TResponse>

A task representing the asynchronous operation, containing the response.

Interface IRequestMiddleware<TRequest, TResponse>

Namespace: [Raftel.Application.Abstractions](#)

Assembly: Raftel.Application.dll

Represents a middleware component in the request pipeline that can intercept and modify the processing of a request before and/or after the next component is invoked.

```
public interface IRequestMiddleware<TRequest, TResponse> where TRequest
: IRequest<TResponse>
```

Type Parameters

TRequest

The type of the request.

TResponse

The type of the response.

Methods

HandleAsync(TRequest, RequestHandlerDelegate<TResponse>)

Processes the request and optionally calls the next delegate in the pipeline.

```
Task<TResponse> HandleAsync(TRequest request, RequestHandlerDelegate<TResponse> next)
```

Parameters


request TRequest

The request instance.

next [RequestHandlerDelegate](#)<TResponse>

The next handler in the pipeline.

Returns

[Task](#)  <TResponse>

A task containing the result of the request.

Interface IRequest<TResponse>

Namespace: [Raftel.Application.Abstractions](#)

Assembly: Raftel.Application.dll

Represents a request that produces a response of type **TResponse**.

```
public interface IRequest<TResponse>
```

Type Parameters

TResponse

The type of response expected from the request.

Class RequestDispatcher

Namespace: [Raftel.Application.Abstractions](#)

Assembly: Raftel.Application.dll

Default implementation of [IRequestDispatcher](#) that resolves the request handler and middleware pipeline from the dependency injection container.

```
public class RequestDispatcher : IRequestDispatcher
```

Inheritance

[object](#) ← RequestDispatcher

Implements

[IRequestDispatcher](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Constructors

RequestDispatcher(IServiceProvider)

Creates a new instance of the [RequestDispatcher](#) class.

```
public RequestDispatcher(IServiceProvider serviceProvider)
```

Parameters

serviceProvider [IServiceProvider](#)

The dependency injection service provider.

Methods

DispatchAsync<TRequest, TResponse>(TRequest)

Dispatches the given request and returns the result asynchronously.

```
public Task<TResponse> DispatchAsync<TRequest, TResponse>(TRequest request) where TRequest  
: IRequest<TResponse>
```

Parameters

request TRequest

The request to dispatch.

Returns

[Task](#)  <TResponse>

The result returned from the request handler or middleware.

Type Parameters

TRequest

The type of the request.

TResponse

The type of the response.

Delegate

RequestHandlerDelegate<TResponse>

Namespace: [Raftel.Application.Abstractions](#)

Assembly: Raftel.Application.dll

Delegate that represents the next handler or middleware in the request processing pipeline.

```
public delegate Task<TResponse> RequestHandlerDelegate<TResponse>()
```

Returns

[Task](#) <TResponse>

The asynchronous result of the request pipeline execution.

Type Parameters

TResponse

The type of response expected.

Namespace Raftel.Application.Commands

Classes

[CommandDispatcher](#)

Default implementation of [ICommandDispatcher](#) that delegates command execution to the [IRequestDispatcher](#) infrastructure.

Interfaces

[ICommand](#)

Represents a command that does not return a value other than a [Result](#).

[ICommandDispatcher](#)

Dispatches a command to its corresponding handler.

[ICommandHandler<TCommand>](#)

Defines a handler for processing a command of type [TCommand](#).

Class CommandDispatcher

Namespace: [Raftel.Application.Commands](#)

Assembly: Raftel.Application.dll

Default implementation of [ICommandDispatcher](#) that delegates command execution to the [IRequestDispatcher](#) infrastructure.

```
public class CommandDispatcher : ICommandDispatcher
```

Inheritance

[object](#) ← CommandDispatcher

Implements

[ICommandDispatcher](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Constructors

CommandDispatcher(IRequestDispatcher)

Initializes a new instance of the [CommandDispatcher](#) class.

```
public CommandDispatcher(IRequestDispatcher dispatcher)
```

Parameters

dispatcher [IRequestDispatcher](#)

The internal dispatcher used to route commands.

Methods

DispatchAsync<TCommand>(TCommand)

Dispatches a command asynchronously.

```
public Task<Result> DispatchAsync<TCommand>(TCommand command) where TCommand : ICommand
```

Parameters

command TCommand

The command instance to process.

Returns

[Task](#) [<Result>](#)

A [Result](#) representing the outcome of the command execution.

Type Parameters

TCommand

The type of command to dispatch.

Interface ICommand

Namespace: [Raftel.Application.Commands](#)

Assembly: Raftel.Application.dll

Represents a command that does not return a value other than a [Result](#).

```
public interface ICommand : IRequest<Result>
```

Interface ICommandDispatcher

Namespace: [Raftel.Application.Commands](#)

Assembly: Raftel.Application.dll

Dispatches a command to its corresponding handler.

```
public interface ICommandDispatcher
```

Methods

DispatchAsync<TCommand>(TCommand)

Dispatches a command asynchronously.

```
Task<Result> DispatchAsync<TCommand>(TCommand command) where TCommand : ICommand
```

Parameters

command TCommand

The command instance to process.

Returns

[Task](#) [Result](#)

A [Result](#) representing the outcome of the command execution.

Type Parameters

TCommand

The type of command to dispatch.

Interface ICommandHandler<TCommand>

Namespace: [Raftel.Application.Commands](#)

Assembly: Raftel.Application.dll

Defines a handler for processing a command of type **TCommand**.

```
public interface ICommandHandler<TCommand> : IRequestHandler<TCommand, Result> where  
TCommand : ICommand
```

Type Parameters

TCommand

The type of command to handle.

Inherited Members

[IRequestHandler<TCommand, Result>.HandleAsync\(TCommand\)](#).

Namespace Raftel.Application.Queries

Classes

[QueryDispatcher](#)

Default implementation of [IQueryDispatcher](#) that delegates query execution to the [IRequestDispatcher](#) infrastructure.

Interfaces

[IQueryDispatcher](#)

Dispatches a query to its corresponding handler and returns the result.

[IQueryHandler<TQuery, TResult>](#)

Defines a handler for processing a query of type `TQuery` that returns a result of type `TResult`.

[IQuery<TResult>](#)

Represents a query that returns a result of type `TResult`.

Interface IQueryDispatcher

Namespace: [Raftel.Application.Queries](#)

Assembly: Raftel.Application.dll

Dispatches a query to its corresponding handler and returns the result.

```
public interface IQueryDispatcher
```

Methods

DispatchAsync<TQuery, TResult>(TQuery)

Dispatches a query asynchronously and retrieves the result.

```
Task<Result<TResult>> DispatchAsync<TQuery, TResult>(TQuery query) where TQuery  
: IQuery<TResult>
```

Parameters

query TQuery

The query instance to process.

Returns

[Task](#) [Result](#) <[Result](#) <TResult>>

A [Result<TValue>](#) representing the outcome of the query execution.

Type Parameters

TQuery

The type of the query to dispatch.

TResult

The type of the result expected from the query.

Interface IQueryHandler<TQuery, TResult>

Namespace: [Raftel.Application.Queries](#)

Assembly: Raftel.Application.dll

Defines a handler for processing a query of type **TQuery** that returns a result of type **TResult**.

```
public interface IQueryHandler<TQuery, TResult> : IRequestHandler<TQuery, Result<TResult>>
where TQuery : IQuery<TResult>
```

Type Parameters

TQuery

The type of the query to handle.

TResult

The type of the result produced by the query.

Inherited Members

[IRequestHandler<TQuery, Result<TResult>>.HandleAsync\(TQuery\)](#).

Interface IQuery<TResult>

Namespace: [Raftel.Application.Queries](#)

Assembly: Raftel.Application.dll

Represents a query that returns a result of type **TResult**.

```
public interface IQuery<TResult> : IRequest<Result<TResult>>
```

Type Parameters

TResult

The type of the result produced by the query.

Class QueryDispatcher

Namespace: [Raftel.Application.Queries](#)

Assembly: Raftel.Application.dll

Default implementation of [IQueryDispatcher](#) that delegates query execution to the [IRequestDispatcher](#) infrastructure.

```
public class QueryDispatcher : IQueryDispatcher
```

Inheritance

[object](#) ← QueryDispatcher

Implements

[IQueryDispatcher](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Constructors

QueryDispatcher(IRequestDispatcher)

Initializes a new instance of the [QueryDispatcher](#) class.

```
public QueryDispatcher(IRequestDispatcher dispatcher)
```

Parameters

dispatcher [IRequestDispatcher](#)

The internal dispatcher used to route queries.

Methods

DispatchAsync<TQuery, TResult>(TQuery)

Dispatches a query asynchronously and retrieves the result.

```
public Task<Result<TResult>> DispatchAsync<TQuery, TResult>(TQuery query) where TQuery  
: IQuery<TResult>
```

Parameters

query TQuery

The query instance to process.

Returns

[Task](#) [Result](#)<TResult>>

A [Result<TValue>](#) representing the outcome of the query execution.

Type Parameters

TQuery

The type of the query to dispatch.

TResult

The type of the result expected from the query.

Namespace Raftel.Domain.Abstractions

Classes

[Error](#)

[Result](#)

[Result<TValue>](#)

Class Error

Namespace: [Raftel.Domain.Abstractions](#)

Assembly: Raftel.Domain.dll

```
public record Error : IEquatable<Error>
```

Inheritance

[object](#) ← Error

Implements

[IEquatable](#) <[Error](#)>

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

Error(string, string)

```
public Error(string Code, string Message)
```

Parameters

Code [string](#)

Message [string](#)

Fields

None

```
public static Error None
```

Field Value

[Error](#)

NullValue

```
public static Error NullValue
```

Field Value

[Error](#)

Properties

Code

```
public string Code { get; init; }
```

Property Value

[string](#)

Message

```
public string Message { get; init; }
```

Property Value

[string](#)

Class Result

Namespace: [Raftel.Domain.Abstractions](#)

Assembly: Raftel.Domain.dll

```
public class Result
```








Inheritance

[object](#)  ← Result

Derived

[Result](#) <TValue>

Inherited Members

[object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  ,
[object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

Constructors

Result(bool, Error)

```
protected Result(bool isSuccess, Error error)
```

Parameters

isSuccess [bool](#) 

error [Error](#)

Properties

Error

```
public Error Error { get; }
```

Property Value

[Error](#)

IsFailure

```
public bool IsFailure { get; }
```

Property Value

[bool](#)

IsSuccess

```
public bool IsSuccess { get; }
```

Property Value

[bool](#)

Methods

Create<TValue>(TValue?)

```
public static Result<TValue> Create<TValue>(TValue? value)
```

Parameters

value TValue

Returns

[Result](#)<TValue>

Type Parameters

TValue

Failure(Error)

```
public static Result Failure(Error error)
```

Parameters

error [Error](#)

Returns

[Result](#)

Failure<TValue>(Error)

```
public static Result<TValue> Failure<TValue>(Error error)
```

Parameters

error [Error](#)

Returns

[Result](#)<TValue>

Type Parameters

TValue

Success()

```
public static Result Success()
```

Returns

[Result](#)

Success<TValue>(TValue)

```
public static Result<TValue> Success<TValue>(TValue value)
```

Parameters

value TValue

Returns

[Result](#)<TValue>

Type Parameters

TValue

Class Result<TValue>

Namespace: [Raftel.Domain.Abstractions](#)

Assembly: Raftel.Domain.dll

```
public class Result<TValue> : Result
```








Type Parameters

TValue

Inheritance

[object](#)  ← [Result](#) ← Result<TValue>

Inherited Members

[Result.IsSuccess](#), [Result.IsFailure](#), [Result.Error](#), [Result.Success\(\)](#), [Result.Success<TValue>\(TValue\)](#), [Result.Failure<TValue>\(Error\)](#), [Result.Create<TValue>\(TValue\)](#), [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Constructors

Result(TValue, bool, Error)

Initializes a new instance of the [Result<TValue>](#) class.

```
protected Result(TValue value, bool isSuccess, Error error)
```

Parameters

value TValue

The value returned by the operation.

isSuccess [bool](#) 

Indicates whether the operation was successful.

error [Error](#)

The error associated with the operation.

Properties

Value

Gets the value returned by the operation, if successful.

```
public TValue Value { get; }
```

Property Value

TValue

Methods

Failure(Error)

Creates a failed result with the specified error.

```
public static Result<TValue> Failure(Error error)
```

Parameters

error [Error](#)

The error associated with the failure.

Returns

[Result](#)<TValue>

A failed result.

Success(TValue)

Creates a successful result with the specified value.

```
public static Result<TValue> Success(TValue value)
```

Parameters

value TValue

The value returned by the operation.

Returns

[Result](#)<TValue>

A successful result containing the value.

Operators

implicit operator Result<TValue>(TValue)

```
public static implicit operator Result<TValue>(TValue value)
```

Parameters

value TValue

Returns

[Result](#)<TValue>

Namespace Raftel.Domain.BaseTypes

Classes

[AggregateRoot<TId>](#)

Represents the root of an aggregate in the domain model.

[Entity<TId>](#)

Base class for all domain entities with a strongly-typed ID.

[TypedGuidId](#)

Base class for identifiers that use [Guid](#) as the backing type.

[TypedId<T>](#)

Represents a strongly-typed identifier based on a primitive type.

Class AggregateRoot<TId>

Namespace: [Raftel.Domain.BaseTypes](#)

Assembly: Raftel.Domain.dll

Represents the root of an aggregate in the domain model.

```
public abstract class AggregateRoot<TId> : Entity<TId> where TId : TypedId<Guid>
```

Type Parameters




TId

The type of the aggregate's identifier.

Inheritance

[object](#)  ← [Entity](#)<TId> ← AggregateRoot<TId>

Inherited Members

[Entity<TId>.Id](#) , [Entity<TId>.Equals\(object\)](#) , [Entity<TId>.GetHashCode\(\)](#) , [object.Equals\(object, object\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

Constructors

AggregateRoot(TId)

Initializes a new instance of the [AggregateRoot<TId>](#) class.

```
protected AggregateRoot(TId id)
```

Parameters

id TId

The identifier of the aggregate root.

Class Entity<TId>

Namespace: [Raftel.Domain.BaseTypes](#)

Assembly: Raftel.Domain.dll

Base class for all domain entities with a strongly-typed ID.


```
public abstract class Entity<TId> where TId : TypedId<Guid>
```

Type Parameters

TId

The type of the entity's identifier.






Inheritance

[object](#)  ← Entity<TId>

Derived

[AggregateRoot<TId>](#)

Inherited Members

[object.Equals\(object, object\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  ,
[object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

Constructors

Entity(TId)

Initializes a new instance of the [Entity<TId>](#) class.

```
protected Entity(TId id)
```

Parameters

id TId

The identifier of the entity.

Properties

Id

Gets the unique identifier of the entity.

```
public TId Id { get; protected set; }
```

Property Value

TId

Methods

Equals(object?)

Determines whether the specified object is equal to the current object.

```
public override bool Equals(object? obj)
```

Parameters

obj [object](#)[↗]

The object to compare with the current object.

Returns

[bool](#)[↗]

[true](#)[↗] if the specified object is equal to the current object; otherwise, [false](#)[↗].

GetHashCode()

Serves as the default hash function.

```
public override int GetHashCode()
```

Returns

[int](#)

A hash code for the current object.

Class TypedGuidId

Namespace: [Raftel.Domain.BaseTypes](#)

Assembly: Raftel.Domain.dll

Base class for identifiers that use [Guid](#) as the backing type.

```
public abstract record TypedGuidId : TypedId<Guid>, IEquatable<TypedId<Guid>>,
    IEquatable<TypedGuidId>
```

Inheritance

[object](#) ← [TypedId](#)<[Guid](#)> ← TypedGuidId

Implements

[IEquatable](#)<[TypedId](#)<[Guid](#)>>, [IEquatable](#)<[TypedGuidId](#)>

Inherited Members

[TypedId](#)<[Guid](#)>.ToString(), [TypedId](#)<[Guid](#)>.Equals([TypedId](#)<[Guid](#)>), [TypedId](#)<[Guid](#)>.GetHashCode(), [object](#).Equals([object](#)), [object](#).Equals([object](#), [object](#)), [object](#).GetType(), [object](#).MemberwiseClone(), [object](#).ReferenceEquals([object](#), [object](#))

Constructors

TypedGuidId(Guid)

Initializes a new instance of the [TypedGuidId](#) class.

```
protected TypedGuidId(Guid value)
```

Parameters

value [Guid](#)

The GUID value.

Methods

NewGuid()

Generates a new GUID using version 7.

```
public static Guid.NewGuid()
```

Returns

[Guid](#)

Class TypedId<T>

Namespace: [Raftel.Domain.BaseTypes](#)

Assembly: Raftel.Domain.dll

Represents a strongly-typed identifier based on a primitive type.

```
public abstract record TypedId<T> : IEquatable<TypedId<T>> where T : notnull
```

Type Parameters



T

The underlying primitive type (e.g., Guid, int).

Inheritance

[object](#)  ← TypedId<T>






Implements

[IEquatable](#)  <[TypedId](#)  <T> >

Derived

[TypedGuidId](#)

Inherited Members

[object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetType\(\)](#)  ,
[object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#) 

Constructors

TypedId(T)

Initializes a new instance of the [TypedId<T>](#) class.

```
protected TypedId(T value)
```

Parameters

value T

The raw value of the identifier.

Exceptions

[ArgumentNullException](#) 

Thrown when value is null.

[ArgumentException](#) 

Thrown when value is Guid.Empty.

Methods

Equals(TypedId<T>?)

Indicates whether the current object is equal to another object of the same type.

```
public virtual bool Equals(TypedId<T>? other)
```



Parameters

other [TypedId<T>](#)

An object to compare with this object.

Returns

[bool](#) 

[true](#)  if the current object is equal to the **other** parameter; otherwise, [false](#) .

GetHashCode()

Serves as the default hash function.

```
public override int GetHashCode()
```

Returns

[int](#)

A hash code for the current object.

ToString()

Returns a string that represents the current object.

```
public override string? ToString()
```

Returns

[string](#)

A string that represents the current object.

Operators

implicit operator T(TypedId<T>)

Implicitly converts a [TypedId<T>](#) to its underlying value.

```
public static implicit operator T(TypedId<T> id)
```

Parameters

id [TypedId<T>](#)

Returns

T