LinksPlatform's Platform.Setters Class Library

./SetterBase[TResult].cs

```csharp
1   using System.Runtime.CompilerServices;
2
3   #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
4
5   namespace Platform.Setters
6   {
7       /// <remarks>
8       /// Must be class, not struct (in order to persist access to Result property value).
9       /// </remarks>
10      public abstract class SetterBase<TResult>
11      {
12          protected TResult _result;
13          public TResult Result => _result;
14          protected SetterBase() { }
15          protected SetterBase(TResult defaultValue) => _result = defaultValue;
16          [MethodImpl(MethodImplOptions.AggressiveInlining)]
17          public void Set(TResult value) => _result = value;
18      }
19  }
```

./Setter[TResult].cs

```csharp
1   using System.Runtime.CompilerServices;
2
3   #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
4
5   namespace Platform.Setters
6   {
7       public class Setter<TResult> : SetterBase<TResult>
8       {
9           public Setter(TResult defaultValue) : base(defaultValue) { }
10
11          public Setter() { }
12
13          [MethodImpl(MethodImplOptions.AggressiveInlining)]
14          public bool SetAndReturnTrue(TResult value)
15          {
16              _result = value;
17              return true;
18          }
19
20          [MethodImpl(MethodImplOptions.AggressiveInlining)]
21          public bool SetAndReturnFalse(TResult value)
22          {
23              _result = value;
24              return false;
25          }
26      }
27  }
```

./Setter[TResult, TDecision].cs

```csharp
1   using System.Collections.Generic;
2   using System.Runtime.CompilerServices;
3
4   #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
5
6   namespace Platform.Setters
7   {
8       public class Setter<TResult, TDecision> : SetterBase<TResult>
9       {
10          private readonly TDecision _trueValue;
11          private readonly TDecision _falseValue;
12
13          public Setter(TDecision trueValue, TDecision falseValue, TResult defaultValue)
14              : base(defaultValue)
15          {
16              _trueValue = trueValue;
17              _falseValue = falseValue;
18          }
19
20          public Setter(TDecision trueValue, TDecision falseValue) : this(trueValue, falseValue,
       ↪  default) { }
21
22          public Setter(TResult defaultValue) : base(defaultValue) { }
23
24          public Setter() { }
25
26          [MethodImpl(MethodImplOptions.AggressiveInlining)]
27          public TDecision SetAndReturnTrue(TResult value)
```

```csharp
        {
            _result = value;
            return _trueValue;
        }

        [MethodImpl(MethodImplOptions.AggressiveInlining)]
        public TDecision SetAndReturnFalse(TResult value)
        {
            _result = value;
            return _falseValue;
        }

        [MethodImpl(MethodImplOptions.AggressiveInlining)]
        public TDecision SetFirstAndReturnTrue(IList<TResult> list)
        {
            _result = list[0];
            return _trueValue;
        }

        [MethodImpl(MethodImplOptions.AggressiveInlining)]
        public TDecision SetFirstAndReturnFalse(IList<TResult> list)
        {
            _result = list[0];
            return _falseValue;
        }
    }
}
```

Index