



Implementacja DDD .NET - Zadanie 2

Bottega IT Minds

1. Zadanie 2 - Value Object

1.1. Wprowadzenie

W Module *Payments* został wprowadzony *ValueObject Money*:

```
public record Money(decimal Amount, string CurrencyCode)
{
    public static Money Of(decimal value, string currencyCode)
    {
        return new Money(value, currencyCode);
    }

    public static Money operator +(Money left, Money right)
    {
        if (left.CurrencyCode != right.CurrencyCode)
        {
            throw new ArgumentException("Currencies don't match.");
        }

        return Of(left.Amount + right.Amount, left.CurrencyCode);
    }

    public static Money operator -(Money left, Money right)
    {
        if (left.CurrencyCode != right.CurrencyCode)
        {
            throw new ArgumentException("Currencies don't match.");
        }

        return Of(left.Amount - right.Amount, left.CurrencyCode);
    }
}
```

Dodatkowo zostały zaimplementowane testy jednostkowe sprawdzające jego zachowanie, poniżej przykład:

```
[Test]
public void GivenMoneyWithTheSameAmountAndCurrency_WhenCompareThem_ThenTheyAreEqual()
{
    // Given
    var firstMoney = Money.Of(100, "PLN");
    var secondMoney = Money.Of(100, "PLN");

    // When
    var areTheyEqual = firstMoney == secondMoney;

    // Then
```

```
areTheyEqual.Should().BeTrue();  
}
```

1.2. Treść Zadania

Na bazie implementacji *Money* w module *Payments* dokonaj refaktoryzacji w module *Availability*. Refaktoryzacja polega na wprowadzaniu *ValueObjectu* **Blockade** który będzie reprezentował blokadę zasobu (**Resource**).

W ramach zadania:

1. Zastanów się, jakie zachowania może mieć blokada i jakie aktualnie atrybuty klasy **Resource** tworzą "konceptyjną całość"
2. Utwórz metodę fabrykującą blokadę tymczasową
3. Utwórz metodę fabrykującą blokadę permanentną
4. Utwórz metodę, która będzie zwracała, czy jest to blokada dla innego właściciela niż ten podany jako parametr.

Dla punktu 4. przetestuj jednostkowo różne przypadki:

- a) Ten sam właściciel
- b) Inny właściciel, blokada permanentna
- c) Inny właściciel, blokada tymczasowo, nie wygasła
- d) Inny właściciel, blokada tymczasowa, wygasła.

Po pomyślnej implementacji VO **Blockade**:

- użyj obiektu w Encji **Resource**
- uruchom wszystkie testy jednostkowe sprawdzając, czy nie zostało przypadkiem zmienione zachowanie.