Alem Muratović

Whitebox testiranje

Testovi su rađeni na metodi calculateWithHolidays u DiscountService:

```
public int calculateWithHolidays(int regularPrice,int numberOfCoins, DateTime date) {
   if (date == null || date < DateTime.Now)
   {
      throw new ArgumentException();
   }
   double holidayDiscount = 0;
   if (date.Month == 1 && date.Day == 1)
   {
      holidayDiscount = 0.5;
   }
   return (int)(regularPrice - (regularPrice * (numberOfCoins / (double)100)) - regularPrice *
holidayDiscount);
}
```

Line coverage:

```
[TestMethod]
public void calculateWithHolidays_dateIsNull_throwsException()
{
   DiscountService discountService = new DiscountService();
   DateTime datum = new DateTime();
   Assert.ThrowsException<ArgumentException>(() => discountService.calculateWithHolidays(100, 50,
datum));
}

[TestMethod]
```

```
public void calculateWithHolidays_invalidDate_throwsException()
{
   DiscountService discountService = new DiscountService();
   DateTime datum = new DateTime(2000, 2, 2);
   Assert.ThrowsException<ArgumentException>(() => discountService.calculateWithHolidays(100, 50,
datum));
}

[TestMethod]
public void calculateWithHolidays_newYearDiscount_returnsDiscountedPrice()
{
   DiscountService discountService = new DiscountService();
   DateTime datum = new DateTime(2025, 1, 1);
   Assert.AreEqual( 0,discountService.calculateWithHolidays(100, 50, datum));
}

[TestMethod]
public void calculateWithHolidays_noHolidayDiscount_returnsDiscountedPrice()
{
   DiscountService discountService = new DiscountService();
   DateTime datum = new DateTime(2025, 2, 1);
   Assert.AreEqual(50, discountService.calculateWithHolidays(100, 50, datum));
}
```