

Zadanie rekrutacyjne zostało zaprojektowane jako aplikacja konsolowa. Początkowo planowane było wykonanie aplikacji webowej z wykorzystaniem frameworka Spring i bazy danych jednak z uwagi na to, że projektowana aplikacja ma wykonywać pojedyncze zadanie obliczania rabatów, rozwiązanie to byłoby niepotrzebnie skomplikowane. Przez wykorzystanie bazy danych ucierpiało by także szybkość działania.

Wykorzystany został wzorzec architektoniczny MVC, który zwiększa elastyczności kodu. Model zawiera klasy prezentujące produkty oraz rabaty w rozumieniu kwoty dla grupy produktów.

Kwota rabatu dla pojedynczego produktu w celu odróżnienia jest nazywana `productDiscount`. Nie jest ona przechowywana w modelu a jedynie wykorzystywana w widoku jako zmienna typu `double`.

Dane wejściowe zaszyte są w kodzie produktu w klasie `ProductController`, dane rabatu w klasie `DiscountController` w metodzie `addBaseData()`.

Rabat dla produktów jest naliczany proporcjonalnie z dokładnością do jednego grosza, a nadmiar przypisywany jest do ostatniego produktu. Odpowiada za to metoda `count()` w klasie `Controller`.

Walidowane są cena produktu (większa równa 0zł) wartość rabatu (większa równa 0zł) i ilość produktów (mniejsza niż 5). W przypadku błędów z produktem nie zostanie on wzięty pod uwagę, natomiast zła wartość rabatu spowoduje ustawienie jej na 0zł.

Cena i rabat zaokrąglane są do wartości 1 gorsza. Cena dodatkowo zaokrąglana jest zawsze w górę (w celu uniknięcia strat przez sklep).