



Uniwersytet Rzeszowski
Kolegium Nauk Przyrodniczych
Instytut Informatyki

Praca projektowa programowanie obiektowe

System obsługi zamówień piekarni

Prowadzący:

mgr inż. Ewa Żesławska

Autor:

Piotr Gajda

nr albumu: 131433

Kierunek: Informatyka, grupa lab 1

Rzeszów 2024

Spis treści

1. Opis założeń projektu	3
2. Specyfikacja wymagań.....	4
3. Opis struktury projektu.....	7
4. Harmonogram realizacji projektu.....	10
5. Prezentacja warstwy użytkowej projektu	12
6. Podsumowanie	17
7. Literatura	18

1. Opis założeń projektu

Cel i założenie projektu

Celem projektu jest opracowanie innowacyjnej aplikacji, która pozwoli na zautomatyzowane i efektywne zarządzanie zamówieniami w piekarni, zapisywanie zamówień oraz generowanie potrzebnych składników do wytworzenia danego produktu.

Problemem, który zostanie rozwiązany, jest nieefektywność i czasochłonność tradycyjnych metod zarządzania zamówieniami piekarni. Podstawowe źródło problemu stanowi brak zintegrowanego systemu, który mógłby usprawnić te procesy.

Problem jest ważny, ponieważ nieefektywne zarządzanie zamówieniami w piekarni prowadzi do marnotrawstwa czasu, zasobów i zmniejsza satysfakcję klientów. Dowody na istnienie problemu obejmują skargi pracowników na długi czas zapisywania zamówienia oraz problemy z błędami w danych klientów.

Aby problem został rozwiązany, niezbędne jest opracowanie i wdrożenie zintegrowanej aplikacji, która będzie obejmować wszystkie kluczowe funkcje dodawanie zamówień. Kluczowe będzie wykorzystanie wiedzy i umiejętności z zakresu programowania, analizy biznesowej oraz bezpieczeństwa danych.

Kroki realizacji projektu:

- Analiza wymagań i projektowanie systemu.
- Opracowanie prototypu aplikacji.
- Testowanie i weryfikacja funkcjonalności aplikacji.

Wynikiem prac będzie w pełni funkcjonalna aplikacja "System Obsługi Zamówień Piekarni", która umożliwi zintegrowane i efektywne zarządzanie zamówieniami, zwiększając tym samym satysfakcję klientów i efektywność pracy piekarni.

2. Specyfikacja wymagań

Niniejsza specyfikacja zawiera szczegółowe wymagania funkcjonalne i нефункционалне, które są fundamentalne dla projektu .

Wymagania funkcjonalne

Poniżej znajduje się lista kluczowych funkcji które powinny być spełnione przez system. Opisują one wszystkie dostępne operacje dla użytkowników aplikacji.

- Przeglądanie i dodawanie zamówień piekarni
 - Umożliwienie użytkownikom zapoznania się z dostępnymi produktami oraz możliwość złożenia zamówienia poszczególnych produktów.
 - Administratorzy i pracownicy systemu powinni móc wyświetlać oraz edytować zamówienia.
- Generowanie składników:
 - Administratorzy powinni mieć dostęp do analizy zamówień firmy, ilości wykupionych produktów.
- Aplikacja desktopowa:
 - Stworzenie kompleksowej aplikacji desktopowej, spełniającej wszystkie założenia projektu.
 - Zapewnienie efektywności, precyzji oraz łatwości dostępu do danych.

Wymagania niefunkcjonalne

Poniżej znajdują się założenia dotyczące jakości i wydajności systemu. Obejmują one aspekty, takie jak bezpieczeństwo, wydajność, dostępność oraz inne właściwości, które wpływają na doświadczenia użytkowników.

- **Użyteczność:**
 - Aplikacja powinna być intuicyjna i łatwa w obsłudze zarówno dla pracowników biura, jak i klientów.
 - Estetyczna i czytelna prezentacja informacji o ofertach, rezerwacjach oraz o użytkownikach.
- **Bezpieczeństwo:**
 - Zapewnienie bezpiecznego przechowywania danych klientów, w tym danych osobowych, oraz informacji na temat zamówień.
- **Wydajność:**
 - Aplikacja powinna działać płynnie bez zbędnych opóźnień.
 - Optymalizacja zamówień do bazy danych w celu zapewnienia płynności działania.
- **Dostępność:**
 - System powinien być dostępny dla użytkowników w dowolnym miejscu i czasie.
 - Aplikacja powinna być dostępna bez błędów na wspieranych systemach operacyjnych.
- **Skalowalność:**
 - Łatwość w dodawaniu nowych funkcji i obsługi powiększającej się liczby użytkowników bez utraty wydajności.
- **Środowisko:**
 - Regularne aktualizacje systemu zgodnie z najnowszymi wersjami bibliotek i środowisk programistycznych.
- **Utrzymanie:**
 - Zastosowanie nowoczesnych narzędzi do zarządzania kodem źródłowym, takich jak systemy kontroli wersji (np. Git), dla efektywnego śledzenia zmian.
 - Zapewnienie dokumentacji technicznej i użytkowej w celu ułatwienia utrzymania i rozwoju systemu.

- Podsumowanie:

Projektowana aplikacja ma na celu usprawnienie procesu składania zamówień, eliminując tradycyjne bariery związane z korzystaniem z aplikacji firmowych. Skupia się na nowoczesnych rozwiązaniach, zapewniając jednocześnie wydajność i intuicyjność obsługi dla użytkowników. System umożliwi łatwe zarządzanie zamówieniami, przeglądanie dostępnych produktów w piekarni oraz monitorowanie wszystkich zamówień, co zwiększy efektywność i satysfakcję klientów i pracowników.

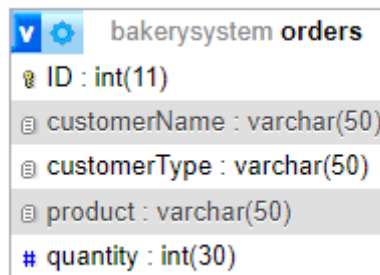
3. Opis struktury projektu

Poniżej przedstawiono informacje dotyczące struktury projektu. Obejmują one używane środowisko programistyczne, narzędzia, minimalne wymagania sprzętowe, hierarchie klas oraz strukturę przechowywanych danych.

- Środowisko programistyczne:
 - Oprogramowanie: IntelliJ IDEA 2024.1.2.
 - Implementacja języka programowania: OpenJDK 17.0.1.
- Narzędzia:
 - Biblioteka graficzna: Swing.
 - Biblioteka mysql-connector-j-8.4.0.
- Rekomendowane wymagania sprzętowe:
 - Procesor: Intel Core i3.
 - Pamięć Ram: 512 MB.
 - Dysk twardy: Minimum 2 GB.
 - System operacyjny: Windows 11.

- Struktura bazy danych:

Na Rys. 1 przedstawiono Diagram ERD bazy danych wymaganej do poprawnego działania aplikacji.



Rys. 1. Diagram ERD bazy danych

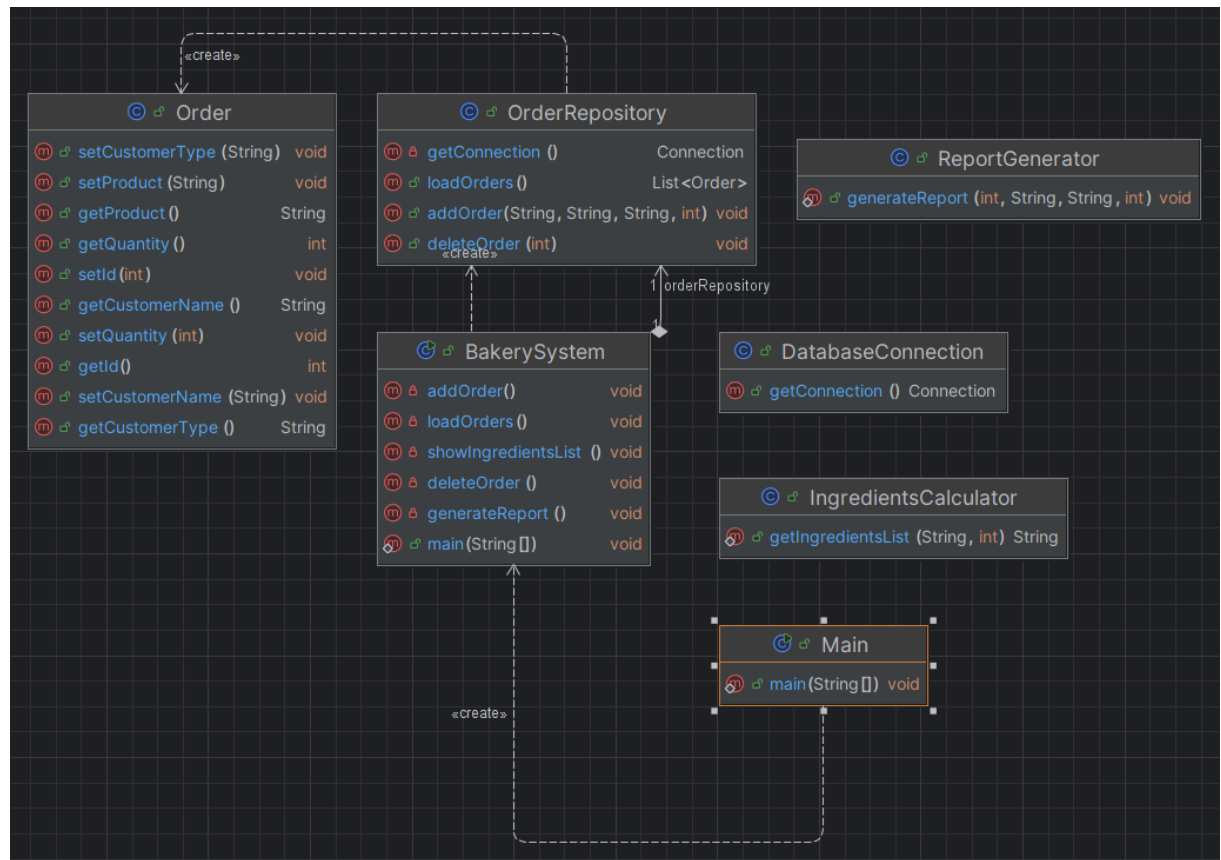
Baza danych składa się z jednej tabeli:

- Tabela „orders” – zawiera informację takie jak:
 - Identyfikator zamówienia – „ID”.
 - Nazwa użytkownika - „customerName”.
 - Typ użytkownika – „customerType”.
 - Rodzaj produktu – „product”.
 - Ilość produktu – „quantity”.

Aplikacja zawiera baze danych w pliku orders.sql, w której zapisują się wszystkie zamówienia piekarni dodane przez pracownika danej piekarni. Przed rozpoczęciem korzystania z aplikacji, wymagane jest zaimportowanie tego pliku do systemu zarządzania bazami danych.

- Hierarchia klas:

Na Rys 2. Przedstawiono Hierarchie klas projektowanej aplikacji.



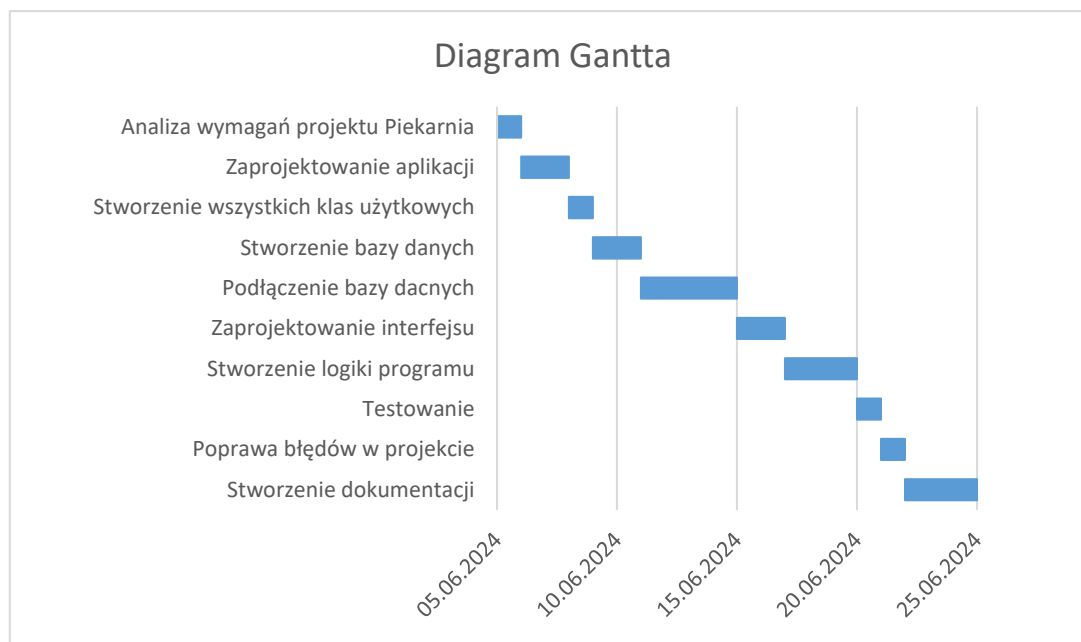
Rys. 2 Hierarchia klas

- Klasa „Main” – pełni rolę głównej klasy programu, umożliwiając uruchomienie aplikacji.
- Klasa „BakerySystem” - stanowi główne okno aplikacji do zarządzania zamówieniami w piekarni. Umożliwia użytkownikowi dodawanie i usuwanie zamówień oraz generowanie faktury i listy potrzebnych składników.
- Klasa „Order” - reprezentuje pojedyncze zamówienie, zawierając informacje o kliencie (CustomerName), produkcie (Produkt) oraz ilości zamówionego produktu (Quantity).
- Klasa „DatabaseConnection” – znajduje się w niej metoda „getConnection”, której celem jest nawiązywanie połączenia z bazą danych.
- Klasa „ReportGenerator” - odpowiada za generowanie faktur. Na podstawie danych z zamówienia
- Klasa „IngredientsCalculator” - oblicza składniki potrzebne do produkcji zamówionych towarów.
- Klasa „OrderRepository” - reprezentuje składniki używane do produkcji produktów. Przechowuje nazwę składnika oraz jego ilość potrzebną do realizacji zamówienia. Jest używana do wyliczania potrzebnych składników na podstawie zamówienia w procesie tworzenia listy składników.

4. Harmonogram realizacji projektu

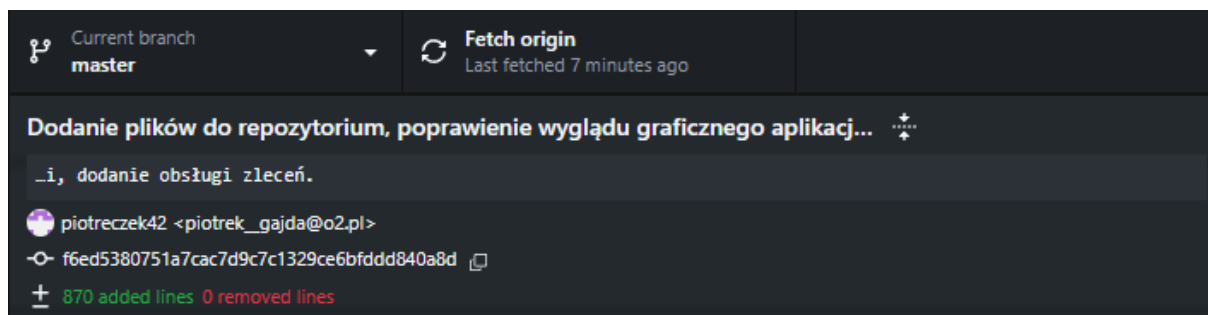
Projekt został zrealizowany w ciągu 20 dni. Poniżej przedstawiono Diagram Gantta(Rys. 3), uwzględniający czas poświęcony na poszczególne etapy projektowania i realizacji projektu.

Diagram Gantta wykonany został w aplikacji Excel.



Rys. 3. Diagram Gantta

Projekt realizowany był z wykorzystaniem systemu kontroli wersji Git, wszystkie pliki źródłowe projektu znajdują się pod adresem: <https://github.com/piotreczek42/Piotr-Gajda-Programowanie-obiektowe> i będą dostępne do 31.12.2024. Na Rys. 4 przedstawiono zrzut ekranu pokazujący historię komitów.



Rys. 4. Historia komitów

5. Prezentacja warstwy użytkowej projektu

Na Rys. 5 przedstawiono Ekran startowy aplikacji, która wyświetla się po otworzeniu programu.

ID	Customer Name	Customer Type	Product	Quantity
9	Piotr	Individual	Bread	12
10	Jakub	Individual	Cake	2
11	Cukiernia "Zosia"	Company	Cookies	50

Rys. 5. Ekran startowy aplikacji

Na Rys. 6 przedstawiono możliwość wyboru rodzaju klienta. Do wyboru są dwie opcje: Individual, Company; to kluczowe do generowania faktury oraz przyznania odpowiedniej zniżki w zależności od wyboru. Firmy dostają 25% zniżkę na wszystkie produkty dostępne w sprzedaży.

ID	Customer Name	Customer Type	Product	Quantity
9	Piotr	Individual	Bread	12
10	Jakub	Individual	Cake	2
11	Cukiernia "Zosia"	Company	Cookies	50

Rys. 6 Wybór rodzaju klienta

Na Rys. 7 przedstawiono możliwość wyboru przez klienta rodzaju produktu. W zależności od wybranej opcji zmienia się ilość składników potrzebnych do wytworzenia produktu oraz cena. Do wyboru są trzy opcje 'Bread', 'Cookies', 'Cake'.

ID	Customer Name	Customer Type	Product	Quantity
9	Piotr	Individual	Bread	12
10	Jakub	Individual	Cake	2
11	Cukiernia "Zosia"	Company	Cookies	50

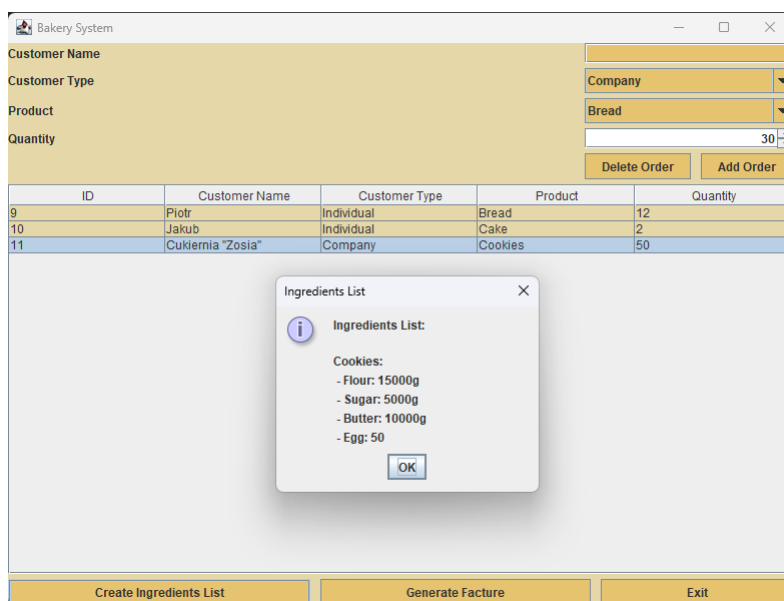
Rys. 7 Wybór rodzaju produktu.

Na Rys. 8 przedstawiono wybór ilości przy składaniu zamówienia. Jest to niezbędne do wygenerowania ceny oraz ilości składników. Podanie ilości zamawianego produktu jest kluczowe przy składaniu zamówienia.

ID	Customer Name	Customer Type	Product	Quantity
9	Piotr	Individual	Bread	12
10	Jakub	Individual	Cake	2
11	Cukiernia "Zosia"	Company	Cookies	50

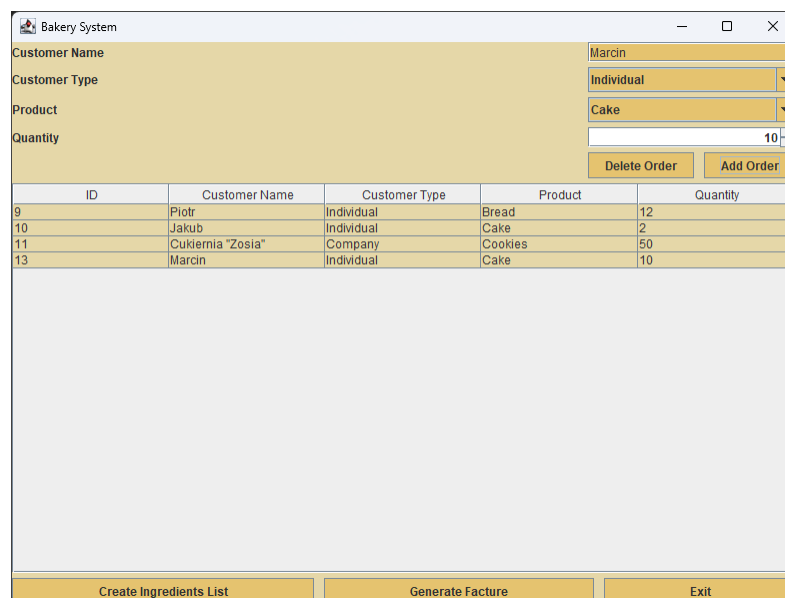
Rys. 8 Wybór ilości

Na Rys. 9 przedstawiono działanie przycisku ‘Create Ingredients List’. Przycisk ten generuje składniki potrzebne do stworzenia produktów, które zostały zamówione. W zależności od produktu, składniki są inne. Przycisk generuje takie wartości jak: ‘Flour’, ‘Sugar’, ‘Butter’, ‘Egg’.



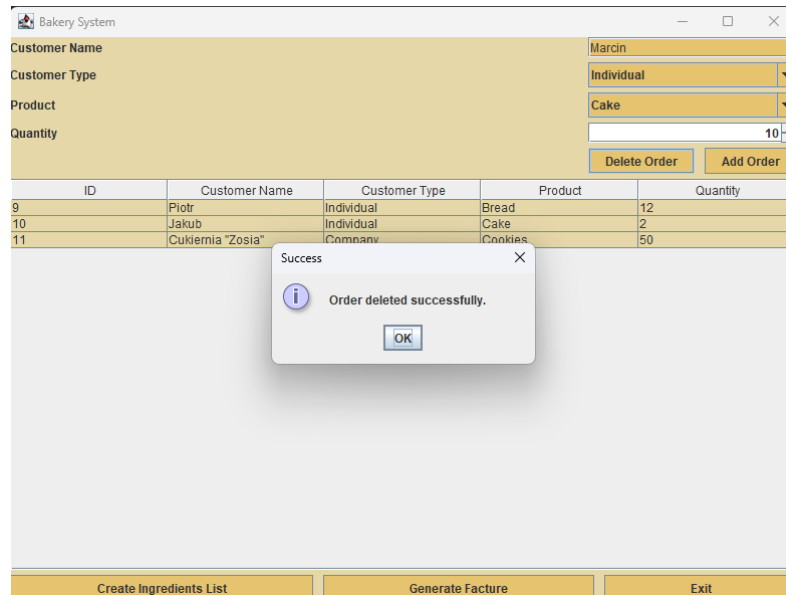
Rys. 9 Generowanie składników

Na Rys. 10 przedstawiono działanie przycisku ‘Add Order’. Przycisk ten dodaje zamówienia do tabeli i bazy danych. Jest to kluczowe w prowadzeniu systemu zamówień piekarni.



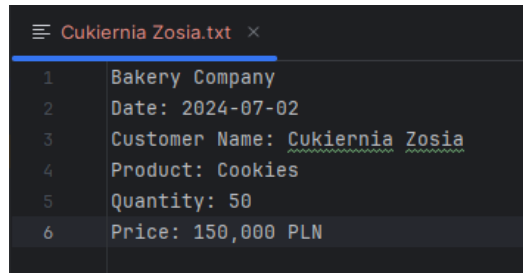
Rys. 10 Dodawanie zamówienia

Na Rys. 11 przedstawiono działanie przycisku 'Delete Order', służy on do usuwania wcześniej dodanego zamówienia. Jest to kluczowe kiedy podczas składania zamówienia znajdzie pomyłka lub jakieś zamówienie zostanie dodane dwa razy.



Rys. 11 Usuwanie zamówienia

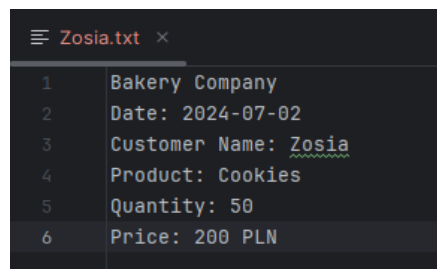
Na Rys. 12 Przedstawiono wygenerowaną fakturę do danego zamówienia. Jest to kluczowe w przypadku, kiedy klient zaznaczył opcję 'Company' w 'Customer Type' ponieważ naliczana jest wtedy zniżka 25%. Na Rys. 13 przedstawiono to samo zamówienie dla indywidualnego klienta



```
≡ Cukiernia Zosia.txt ×
1 Bakery Company
2 Date: 2024-07-02
3 Customer Name: Cukiernia Zosia
4 Product: Cookies
5 Quantity: 50
6 Price: 150,000 PLN
```

1	Bakery Company
2	Date: 2024-07-02
3	Customer Name: Cukiernia Zosia
4	Product: Cookies
5	Quantity: 50
6	Price: 150,000 PLN

Rys. 12 Wygenerowana faktura ze zniżką



```
≡ Zosia.txt ×
1 Bakery Company
2 Date: 2024-07-02
3 Customer Name: Zosia
4 Product: Cookies
5 Quantity: 50
6 Price: 200 PLN
```

1	Bakery Company
2	Date: 2024-07-02
3	Customer Name: Zosia
4	Product: Cookies
5	Quantity: 50
6	Price: 200 PLN

Rys. 13 Wygenerowana faktura bez zniżki

6. Podsumowanie

Realizacja projektu „Piekarnia” zakończyła się sukcesem, spełniając wszystkie cele i założenia. Stworzony system skutecznie wspiera pracowników piekarni w zarządzaniu zamówieniami. Projekt posiada solidne fundamenty, które można rozbudować o dodatkowe funkcje, takie jak reklamy, zniżki dla stałych klientów, nowe produkty oraz menu dla pracowników z możliwością dodawania pozycji.

Te rozszerzenia znacząco podniosą funkcjonalność i atrakcyjność programu, oferując użytkownikom szerszy wachlarz możliwości. Proponowane modyfikacje i udoskonalenia mogą jeszcze bardziej dopasować system do potrzeb użytkowników, czyniąc go jeszcze bardziej użytecznym i efektywnym narzędziem dla piekarni.

7. Literatura

1. Jerzy Krawiec, JAVA. Programowanie obiektowe w praktyce, 2018, Wyd. Oficyna Wydawnicza Politechniki Warszawskiej
2. <https://javastart.pl/baza-wiedzy/programowanie-obiektowe/wyjatki-blok-try-catch> (data dostępu: 07.06.2024)
3. <https://www.samouczekprogramisty.pl/wyjatki-w-jezyku-java/> (data dostępu: 10.06.2024)
4. <https://javappa.com/kurs-wzorce-projektowe/singleton> (data dostępu: 15.06.2024)
5. <https://www.jfree.org/jfreechart/> (data dostępu: 19.06.2024)
6. <https://mvnrepository.com/artifact/org.jfree/jcommon/1.0.23> (data dostępu: 29.06.2024)
7. <https://www.samouczekprogramisty.pl/wyjatki-w-jezyku-java/> (data dostępu 29.06.2024)
8. <https://www.samouczekprogramisty.pl/wyjatki-w-jezyku-java/> (data dostępu 30.06.2024)