

Dokumentacja Projektu

Bazy Danych 1

Natalia Przetocka

24 stycznia 2023

1 Projekt koncepcji, założenia

1.1 Temat projektu

Projekt przedstawia program zarządzający siecią sklepów.

1.2 Analiza wymagań użytkownika

Baza danych zawiera informacje na temat działania sklepów, czyli informacje o pracownikach i zaopatrzeniu sklepów oraz magazynów, a także dane klientów, i historię ich zamówień. Każdy użytkownik może sprawdzić zawartość bazy danych, jednak tylko zalogowany pracownik może dodać rekordy do bazy (może dodać sklepy, magazyny, produkty, pracowników, zaopatrzenie do sklepów, zaopatrzenie do magazynów i może złożyć zamówienie dla klienta).

1.3 Funkcje realizowane w bazie danych

System pozwala niezalogowanemu pracownikowi na przeglądanie danych zawartych w bazie za pomocą raportów prostych jak i widoków. Niektóre widoki wykorzystują funkcje agregujące pozwalające na przegląd podstawowych danych statystycznych zamówień. Po zalogowaniu pracownik ma możliwość dodawania rekordów do bazy. Przy składaniu zamówienia uruchamia się wyzwalacz, który automatycznie usuwa odpowiednie produkty z zaopatrzenia magazynu.

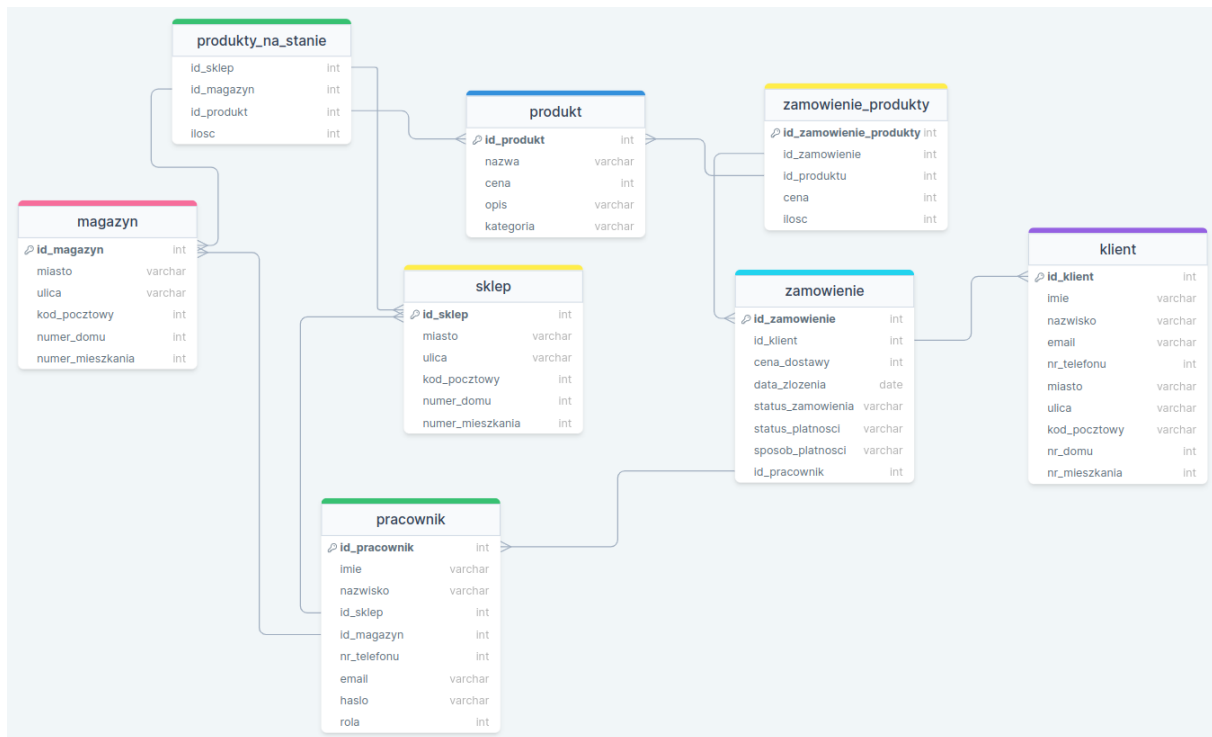
2 Projekt diagramów

2.1 Encje oraz ich atrybuty

W bazie danych występują następujące encje:

- sklep - id_sklep, miasto, ulica, kod_pocztowy, nr_domu, nr_mieszkania,
- magazyn - id_magazyn, miasto, ulica, kod_pocztowy, nr_domu, nr_mieszkania,
- pracownik - id_pracownik, imie, nazwisko, id_sklep, id_magazyn, nr_telefonu, email, rola,
- produkt - id_produkt, cena, nazwa, opis, kategoria,
- produkty_na_stanie - id_sklep, id_magazyn, id_produkt, ilosc,
- zamowienie_produkty - id_zamowienie_produkty, id_zamowienie, id_produkt, cena, ilosc,
- zamowienie - id_zamowienie, id_klient, cena_dostawy, data_zlozenia, status_zamowienia, sposob_platnosci, status_platnosci, id_pracownik,
- klient - id_klient, imie, nazwisko, email, nr_telefonu, miasto, ulica, kod_pocztowy, nr_domu, nr_mieszkania.

2.2 Relacje pomiędzy encjami



Rysunek 1: Diagram ERD sklepu.

3 Projekt logiczny

3.1 Projektowanie tabel, kluczy, indeksów

3.1.1 Encja sklep

Encja sklep - reprezentuje pojedynczy sklep:

1. id_sklep - klucz główny, unikatowa wartości typu INTEGER,
2. miasto - atrybut reprezentujący miasto w którym znajduje się sklep, typu VARCHAR(50),
3. ulica - atrybut reprezentujący ulicę, przy której znajduje się sklep, typu VARCHAR(50),
4. kod_pocztowy - atrybut reprezentujący kod pocztowy miejsca, w którym znajduje się sklep, typu VARCHAR(50),
5. nr_domu - atrybut reprezentujący numer budynku, w którym znajduje się sklep, typu INTEGER,
6. nr_mieszkania - atrybut reprezentujący opcjonalny numer lokalu, w którym znajduje się sklep, typu INTEGER.

3.1.2 Encja magazyn

Encja magazyn - reprezentuje pojedynczy magazyn:

1. id_magazyn - klucz główny, unikatowa wartości typu INTEGER,
2. miasto - atrybut reprezentujący miasto w którym znajduje się magazyn, typu VARCHAR(50),
3. ulica - atrybut reprezentujący ulicę, przy której znajduje się magazyn, typu VARCHAR(50),
4. kod_pocztowy - atrybut reprezentujący kod pocztowy miejsca, w którym znajduje się magazyn, typu VARCHAR(50),

5. nr_domu - atrybut reprezentujący numer budynku, w którym znajduje się magazyn, typu INTEGER,
6. nr_mieszkania - atrybut reprezentujący opcjonalny numer lokalu, w którym znajduje się magazyn, typu INTEGER.

3.1.3 Encja pracownik

Encja pracownik - reprezentuje pojedynczego pracownika:

1. id_pracownik - klucz główny, unikatowa wartości typu INTEGER,
2. imie - atrybut reprezentujący imię pracownika, typu VARCHAR(50),
3. nazwisko - atrybut reprezentujący nazwisko pracownika, typu VARCHAR(50),
4. id_sklep - klucz obcy typu INTEGER, id_sklepu, w którym pracuje pracownik,
5. id_magazyn - klucz obcy typu INTEGER, id_magazynu, w którym pracuje pracownik,
6. nr_telefonu - atrybut reprezentujący numer telefonu pracownika, typu INTEGER,
7. email - atrybut reprezentujący email pracownika, typu VARCHAR(50),
8. haslo - atrybut reprezentujący hasło pracownika, typu VARCHAR(50),
9. rola - atrybut opisujący rolę pracownika (pracownik magazynu, pracownik sklepu itp), typu INTEGER.

3.1.4 Encja produkt

Encja produkt - reprezentuje pojedynczy produkt:

1. id_produkt - klucz główny, unikatowa wartości typu INTEGER,
2. cena - atrybut reprezentujący cenę produktu, typu FLOAT,
3. nazwa - atrybut opisujący nazwę produktu, typu VARCHAR(50),
4. opis - atrybut opisujący szczegóły danego produktu (skład, materiał wykonania itp.), typu VARCHAR(50),
5. kategoria - atrybut opisujący kategorię w jakiej znajduje się produkt, typu VARCHAR(5).

3.1.5 Encja produkty_na_stanie

Encja produkty_na_stanie - reprezentuje produkty znajdujące się w danym sklepie lub magazynie:

1. id_sklep - klucz obcy odpowiadający sklepowi, w którym znajduje się dany produkt, typu INTEGER,
2. id_magazyn - klucz obcy odpowiadający magazynowi, w którym znajduje się dany produkt, typu INTEGER,
3. id_produkt - klucz obcy odpowiadający produktowi, który znajduje się w sklepie bądź magazynie, typu INTEGER,
4. ilosc - atrybut opisujący ilość produktu w danym sklepie lub magazynie, typu INTEGER.

3.1.6 Encja zamowienie_produkty

Encja zamowienie_produkty - reprezentuje produkty z danego zamówienia:

1. id_zamowienie_produkty - klucz główny, unikatowa wartość typu INTEGER,
2. id_zamowienie - klucz obcy, opisujący w którym zamówieniu znajduje się dany produkt, typu INTEGER
3. id_produkt - klucz obcy opisujący produkt w zamówieniu, typu INTEGER,
4. cena - atrybut opisujący cenę jednego produktu, typ FLOAT,
5. ilość - atrybut opisujący ilość produktów w zamówieniu, typ INTEGER.

3.1.7 Encja zamówienie

Encja zamówienie - reprezentuje pojedyncze zamówienia:

1. id_zamowienie - klucz główny, unikatowa wartość typu INTEGER,
2. id_klient - klucz obcy, reprezentuje klienta, któremu przypisane jest dane zamówienie, typu INTEGER,
3. cena_dostawy - atrybut opisujący cenę dostawy zamówienia, typu FLOAT,
4. data_zlozenia - data złożenia zamówienia, typu DATE,
5. staus_zamowienia - atrybut opisujący status zamówienia (złożone, opłacone, wysłane itp.), typu VARCHAR(50),
6. sposob_platnosci - atrybut opisujący sposób płatności (gotówką, blik itp.), typu VARCHAR(50),
7. status_platnosci - atrybut opisujący status płatności (opłaconie, nie opłacone), typu VARCHAR(50),
8. id_pracownik - klucz obcy opisujący pracownika, który złożył zamówienie, typu INTEGER.

3.1.8 Encja klient

Encja klient - reprezentuje pojedynczego klienta:

1. id_klient - klucz główny, unikatowa wartości typu INTEGER,
2. imie - atrybut reprezentujący imię pracownika, typu VARCHAR(50),
3. nazwisko - atrybut reprezentujący nazwisko pracownika, typu VARCHAR(50),
4. email - atrybut reprezentujący email klienta, typu VARCHAR(50),
5. nr_telefonu - atrybut reprezentujący numer telefonu klienta, typu INTEGER,
6. miasto - atrybut reprezentujący miasto do dostawy dla danego klienta, typu VARCHAR(50),
7. ulica - atrybut reprezentujący ulicę do dostawy dla danego klienta, typu VARCHAR(50),
8. kod_pocztowy - atrybut reprezentujący kod pocztowy do dostawy dla danego klienta, typu VARCHAR(50),
9. nr_domu - atrybut reprezentujący numer budynku do dostawy dla danego klienta, typu INTEGER,
10. nr_mieszkania - atrybut reprezentujący opcjonalny numer lokalu do dostawy dla danego klienta, typu INTEGER.

3.2 Słownik danych

Tabela 1: Encja sklep

Atrybut	Typ	Ograniczenie	Klucz
id_sklep	INTEGER	NOT NULL	PK
miasto	VARCHAR(50)	NOT NULL	-
ulica	VARCHAR(50)	NOT NULL	-
kod_pocztowy	VARCHAR(50)	NOT NULL	-
nr_domu	VARCHAR(50)	NOT NULL	-
nr_mieszkania	VARCHAR(50)	-	-

Tabela 2: Encja magazyn

Atrybut	Typ	Ograniczenie	Klucz
id_magazyn	INTEGER	NOT NULL	PK
miasto	VARCHAR(50)	NOT NULL	-
ulica	VARCHAR(50)	NOT NULL	-
kod_pocztowy	VARCHAR(50)	NOT NULL	-
nr_domu	VARCHAR(50)	NOT NULL	-
nr_mieszkania	VARCHAR(50)	-	-

Tabela 3: Encja pracownik

Atrybut	Typ	Ograniczenie	Klucz
id_pracownik	INTEGER	NOT NULL	PK
imie	VARCHAR(50)	NOT NULL	-
nazwisko	VARCHAR(50)	NOT NULL	-
id_sklep	INTEGER	-	FK dla sklep
id_magazyn	INTEGER	-	FK dla magazyn
nr_telefonu	VARCHAR(50)	NOT NULL	-
email	VARCHAR(50)	NOT NULL	-
haslo	VARCHAR(50)	NOT NULL	-
rola	INTEGER	NOT NULL	-

Tabela 4: Encja produkt

Atrybut	Typ	Ograniczenie	Klucz
id_produkt	INTEGER	NOT NULL	PK
cena	FLOAT	NOT NULL	-
nazwa	VARCHAR(50)	NOT NULL	-
opis	VARCHAR(50)	NOT NULL	-
kategoria	VARCHAR(50)	NOT NULL	-

Tabela 5: Encja produkty_na_stanie

Atrybut	Typ	Ograniczenie	Klucz
id_sklep	INTEGER	-	FK dla sklep
id_magazyn	INTEGER	-	FK dla magazyn
id_produkt	INTEGER	NOT NULL	FK dla produkt
ilosc	INTEGER	NOT NULL	-

Tabela 6: Encja zamowienie_produkty

Atrybut	Typ	Ograniczenie	Klucz
id_zamowienie_produkty	INTEGER	NOT NULL	PK
id_zamowienie	INTEGER	NOT NULL	FK dla zamowienie
id_produkt	INTEGER	NOT NULL	FK dla produkt
cena	FLOAT	NOT NULL	-
ilosc	INTEGER	NOT NULL	-

Tabela 7: Encja zamowienie

Atrybut	Typ	Ograniczenie	Klucz
id_zamowienie	INTEGER	NOT NULL	PK
id_klient	INTEGER	NOT NULL	FK dla klient
cena_dostawy	FLOAT	NOT NULL	-
data_zlozenia	DATA	NOT NULL	-
status_zamowienia	VARCAHR(50)	NOT NULL	-
sposob_platnosci	VARCAHR(50)	NOT NULL	-
status_platnosci	VARCAHR(50)	NOT NULL	-
is_pracownik	INTEGER	-	FK dla pracownik

Tabela 8: Encja zamowienie

Atrybut	Typ	Ograniczenie	Klucz
id_klient	INTEGER	NOT NULL	PK
imie	VARCHAR(50)	NOT NULL	-
nazwisko	VARCHAR(50)	NOT NULL	-
email	VARCHAR(50)	NOT NULL	-
nr_telefonu	VARCHAR(50)	NOT NULL	-
miasto	VARCHAR(50)	NOT NULL	-
ulica	VARCHAR(50)	NOT NULL	-
kod_pocztowy	VARCHAR(50)	NOT NULL	-
nr_domu	VARCHAR(50)	NOT NULL	-
nr_mieszkania	VARCHAR(50)	-	-

3.3 Zaprojektowanie operacji na danych

Pracownik może wprowadzać dane do każdej tabeli poprzez formularze. Wyświetlanie części tabel odbywa się poprzez widoki:

- pracownicy magazynów - id oraz adres magazynu, imię, nazwisko i dane kontaktowe pracownika, wraz z jego rolą,
- pracownicy sklepów - id oraz adres sklepu, imię, nazwisko i dane kontaktowe pracownika, wraz z jego rolą,
- produkty w magazynach - id i nazwa produktu, jego ilość w magazynie oraz id a także adres magazynu,
- produkty w sklepach - id i nazwa produktu, jego ilość w sklepie oraz id a także adres sklepu,
- zarys zamówień - dane klienta, koszt produktów i dostawy oraz status zamówienia,
- produkty w zamówieniach - dane klienta, nazwa, opis i ilość produktu oraz koszt pojedynczego i wszystkich produktów o danym id,
- zamówienia klientów - dane klienta, jego najczęściej zamawiany produkt (jego nazwa, opis i ile egzemplarzy łącznie było zamówione), łączna ilość zamówień.

Dodatkowo w celu sprawdzenia danych wprowadzanych do bazy wykorzystano CHECK'i:

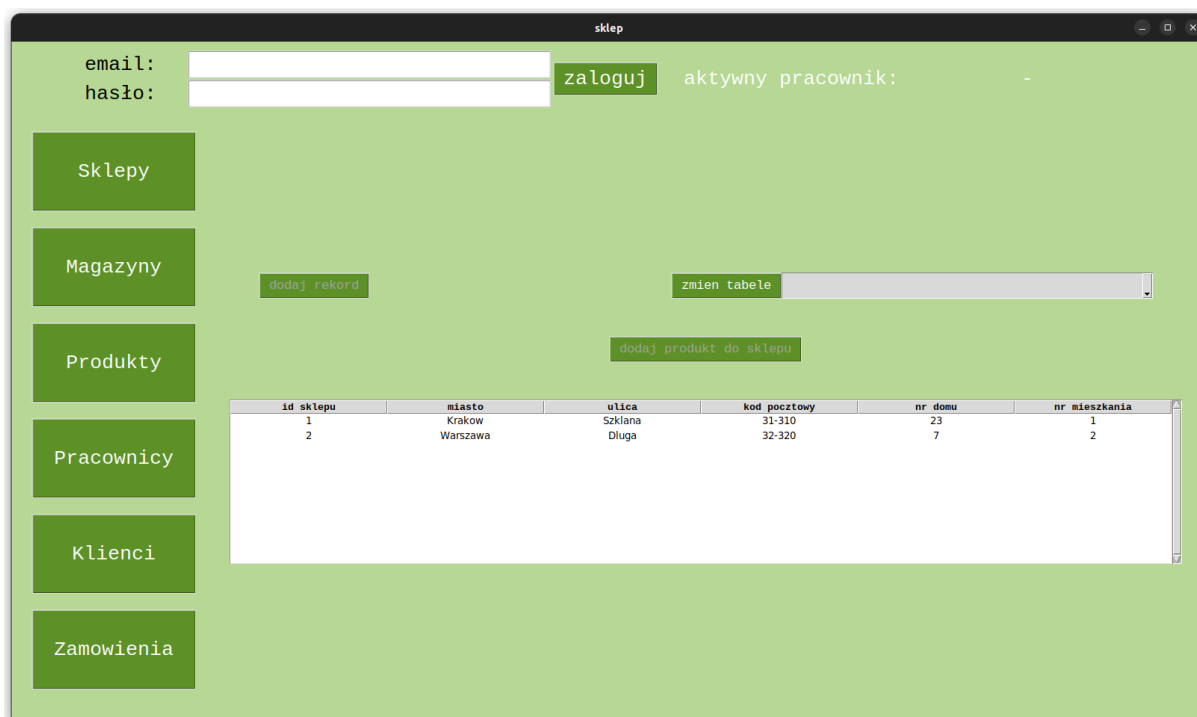
- sprawdzenie czy kod pocztowy pasuje do wzoru `'**-***'`, gdzie gwiazdka to dowolna cyfra,
- sprawdzenie czy email składa się ze słowa, które zaczyna się na literę, następnie jest w nim '@' i subdomeny,
- rola pracownika musi zawierać się w wartościach 1-2, jeśli pracuje w sklepie (id_magazyn jest puste a id_sklep nie jest puste) i w wartościach 3-4 jeśli pracuje w magazynie (id_sklep jest puste, a id_magazyn nie jest puste),
- nr telefonu jest liczbą w przedziale $[10^8, 10^9)$.

4 Projekt funkcjonalny

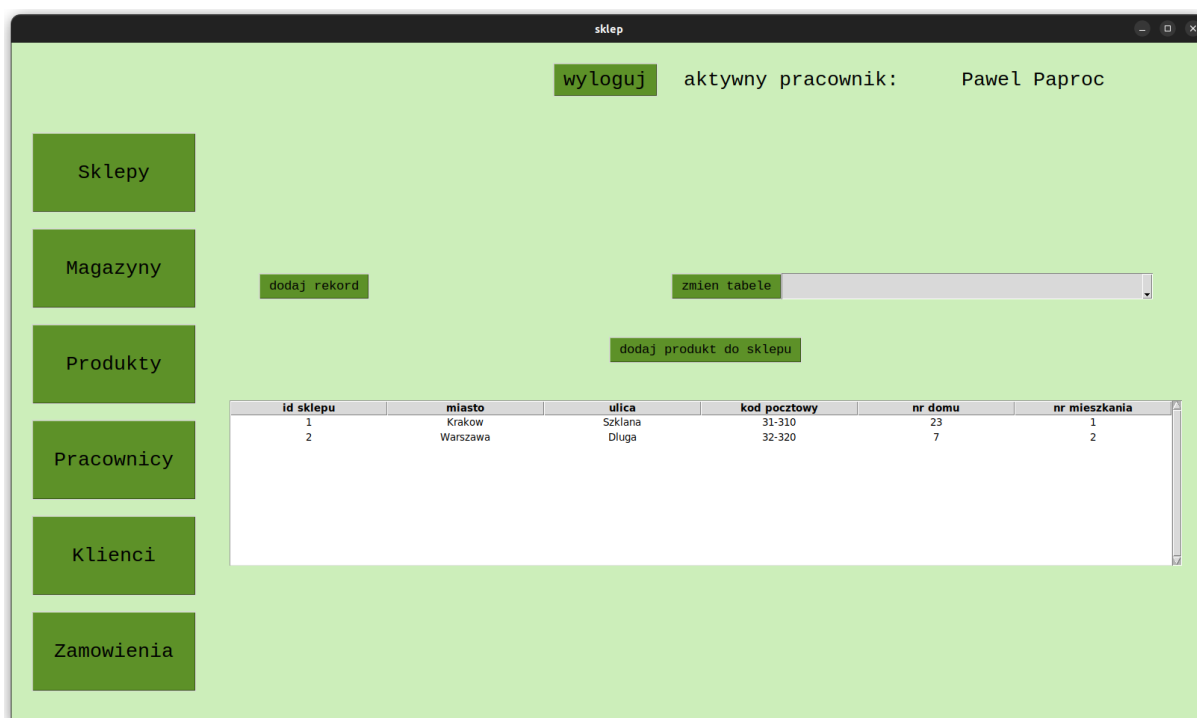
Po włączeniu aplikacji pojawia się 'Strona główna' (Rysunek 2). Na górze widoczny jest 'panel logowania' natomiast po lewej stronie znajduje się 'panel sterowania aplikacją' w postaci przycisków zmieniających wyświetlane widoki lub możliwe dodawanie rekordów (*Sklepy, Magazyny, Produkty, Pracownicy, Klienci, Zamowienia*).

Domyślnie dane w aplikację dotyczą tabeli sklep. Za pomocą przycisków po lewej stronie okna, można zmienić tabelę, której będą dotyczyły raporty i formularze wprowadzające dane.

W celu uzyskania możliwości dodawania rekordów do bazy należy zalogować się. Po zalogowaniu interfejs wygląda jak na rysunku 3.



Rysunek 2: Strona główna.



Rysunek 3: Strona główna po zalogowaniu.

Następnie możemy otworzyć formularz wprowadzający dane do tabeli sklep (rysunek 4) lub do tabeli produkty_na_stanie (rysunek 5) albo wyświetlić dodatkowe raporty związane z tabelą sklep (rysunki 6 i 7).

formularz

id sklepu

miasto

ulica

kod pocztowy

nr domu

nr mieszkania

dodaj zamknij

Rysunek 4: Formularz dodający sklep (wprowadzanie danych do tabeli sklep).

dodaj produkt do sklepu

włóczka {100% bawełna, biała} dodaj

włóczka {100% bawełna, czarna} dodaj

włóczka {100% bawełna, zielona} dodaj

włóczka {100% bawełna, czerwona} dodaj

włóczka {100% bawełna, żółta} dodaj

włóczka {100% kaszmir, biała} dodaj

włóczka {100% kaszmir, zielona} dodaj

włóczka {100% kaszmir, niebieska} dodaj

włóczka {100% kaszmir, fioletowa} dodaj

włóczka {100% kaszmir, różowa} dodaj

szydełko {metalowe, 3mm} dodaj

szydełko {metalowe, 5mm} dodaj

szydełko {metalowe, 6mm} dodaj

szydełko {plastikowe, 7mm} dodaj

szydełko {plastikowe, 8mm} dodaj

zamknij

Rysunek 5: Formularz dodający zaopatrzenie sklepów (wprowadzanie danych do tabeli produkty_na_stanie).

sklep

wyloguj aktywny pracownik: Paweł Paproc

Sklepy

Magazyny

Produkty

Pracownicy

Klienci

Zamowienia

dodaj rekord

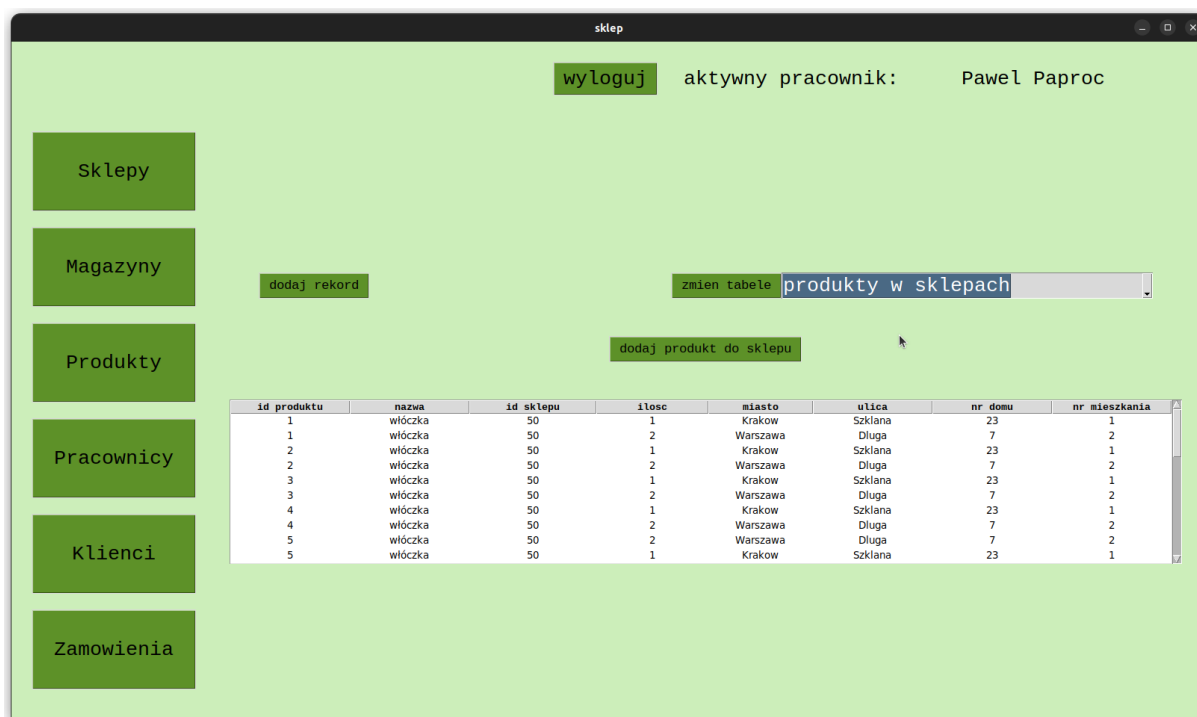
zmien tabela

dodaj produkt do sklepu

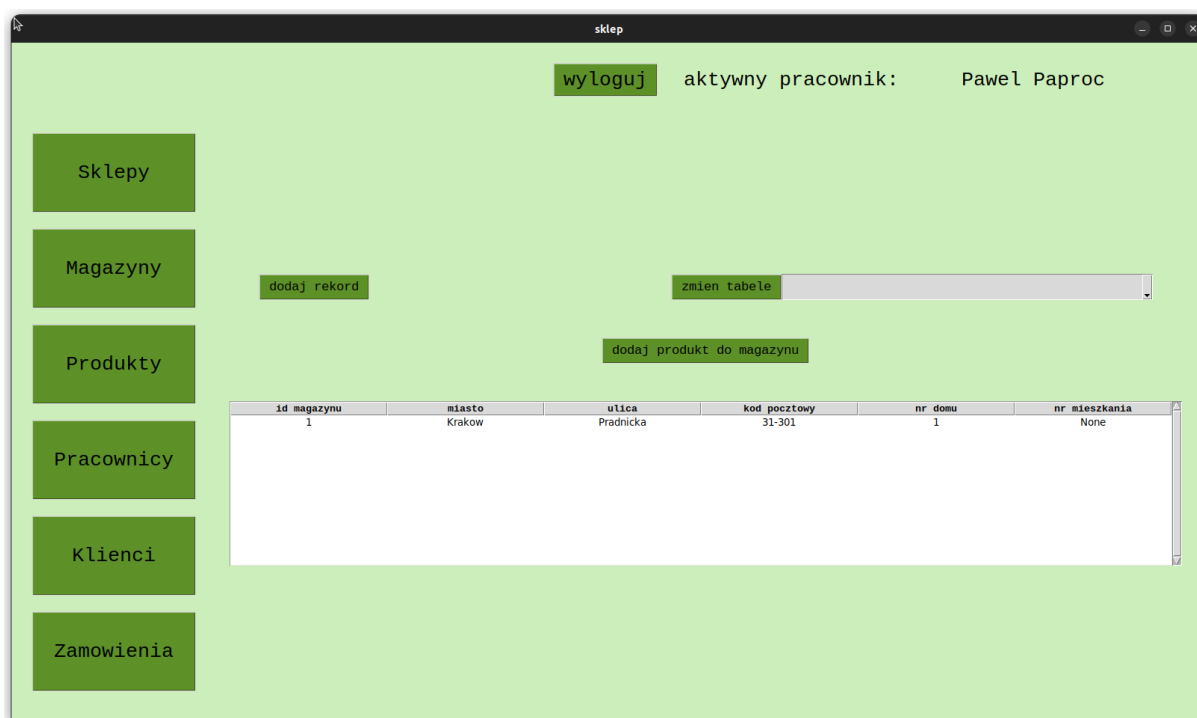
id sklepu	miasto	ulica	nr domu	nr mieszkania	imie	nazwisko	email	nr telefonu	rola
1	Krakow	Szklana	23	1	Paweł	Paproc	PawełPaproc@mail.pl	769795420	kierownik sklepu
2	Warszawa	Długa	7	2	Daniel	Dabrowski	DanielDabrowski@mail.pl	477918857	pracownik sklepu
1	Krakow	Szklana	23	1	Tomasz	Tortownica	TomaszTortownica@mail.pl	629819989	pracownik sklepu
2	Warszawa	Długa	7	2	Czesław	Krzesło	CzesławKrzesło@mail.pl	801811969	pracownik sklepu

Rysunek 6: Raport oparty na widoku, pokazuje pracowników sklepów.

Po kliknięciu przycisku 'Magazyny' widzimy interfejs 8.



Rysunek 7: Raport oparty na widoku, pokazuje produkty w sklepach.



Rysunek 8: Zakładka magazyny.

Następnie możemy otworzyć formularz wprowadzający dane do tabeli magazyn (rysunek 9) lub do tabeli produkty_na_stanie (rysunek 10) albo wyświetlić widoki związane z tabelą magazyn (rysunki 11 i 12).

formularz

id magazynu

miasto

ulica

kod pocztowy

nr domu

nr mieszkania

dodaj zamknij

Rysunek 9: Formularz dodający magazyn (wprowadzanie danych do tabeli magazyn).

dodaj produkt do sklepu

włóczka {100% bawełna, biała} dodaj

włóczka {100% bawełna, czarna} dodaj

włóczka {100% bawełna, zielona} dodaj

włóczka {100% bawełna, czerwona} dodaj

włóczka {100% bawełna, żółta} dodaj

włóczka {100% kaszmir, biała} dodaj

włóczka {100% kaszmir, zielona} dodaj

włóczka {100% kaszmir, niebieska} dodaj

włóczka {100% kaszmir, fioletowa} dodaj

włóczka {100% kaszmir, różowa} dodaj

szydełko {metalowe, 3mm} dodaj

szydełko {metalowe, 5mm} dodaj

szydełko {metalowe, 6mm} dodaj

szydełko {plastikowe, 7mm} dodaj

szydełko {plastikowe, 8mm} dodaj

zamknij

Rysunek 10: Formularz dodający zaopatrzenie magazynów (wprowadzanie danych do tabeli produkty_na_stanie).

sklep

wyloguj aktywny pracownik: Paweł Paproc

Sklepy

Magazyny

Produkty

Pracownicy

Klienci

Zamowienia

dodaj rekord

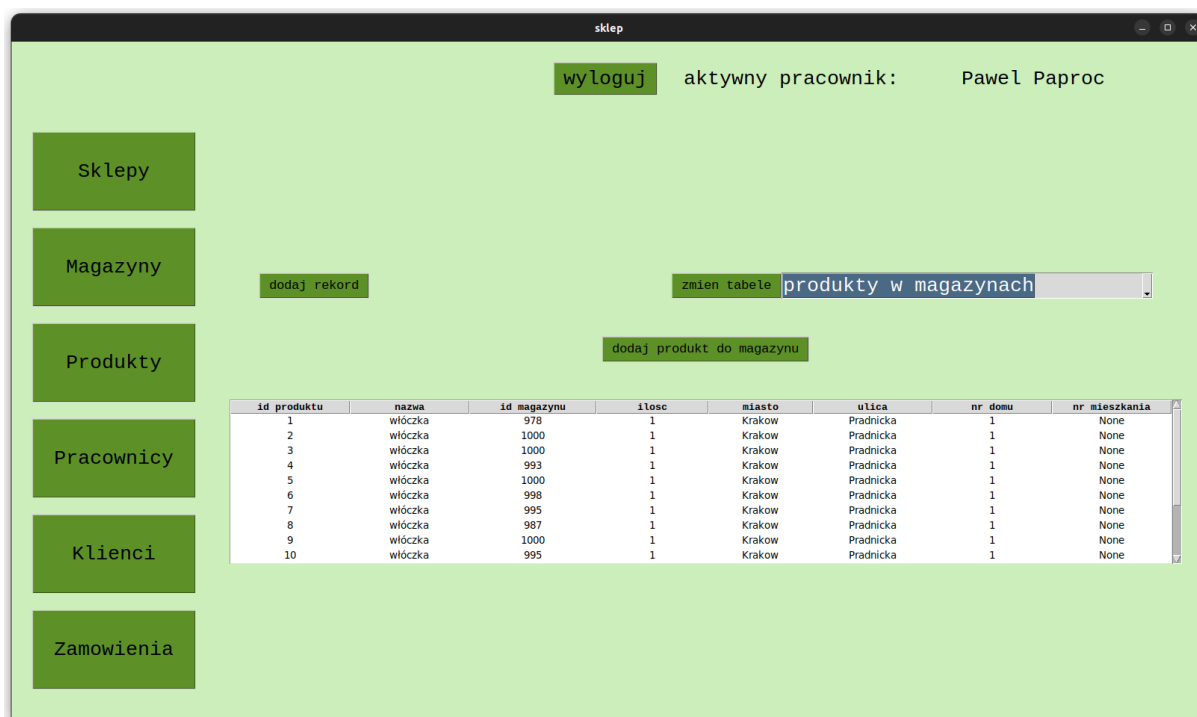
zmien tabelę pracownicy magazynow

dodaj produkt do magazynu

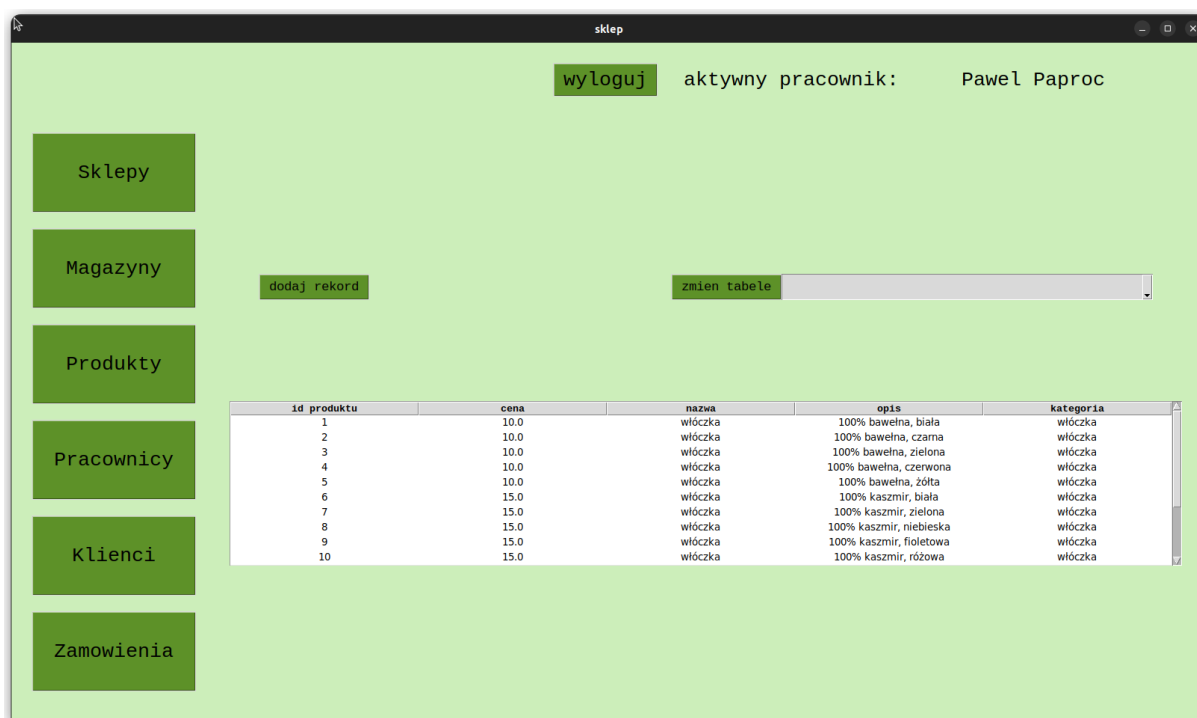
id magazynu	miasto	ulica	kod pocztowy	nr domu	nr mieszkania
1	Krakow	Pradnicka	31-301	1	None

Rysunek 11: Raport oparty na widoku, pokazuje pracowników magazynów.

Po kliknięciu przycisku 'Produkty' widzimy interfejs [13](#).



Rysunek 12: Raport oparty na widoku, pokazuje produkty w magazynach.



Rysunek 13: Zakładka produkty.

Następnie możemy otworzyć formularz wprowadzający dane do tabeli produkt (rysunek 14) albo wyświetlić widoki związane z tabelą produkt (rysunki 15 i 16).

formularz

id produktu

cena

nazwa

opis

kategoria

dodaj zamknij

Rysunek 14: Formularz dodający produkt (wprowadzanie danych do tabeli produkt).

sklep

wyloguj aktywny pracownik: Pawel Paproc

Sklepy

Magazyny

Produkty

Pracownicy

Klienci

Zamowienia

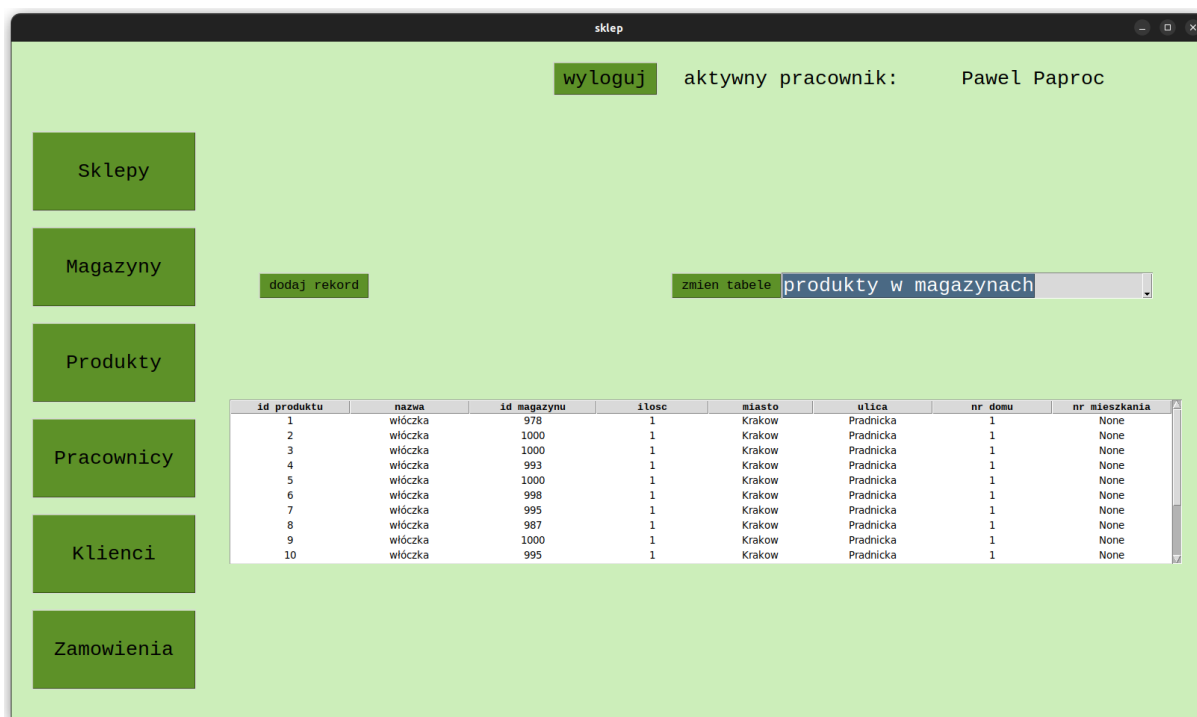
dodaj rekord

zmien tabele produkty w sklepach

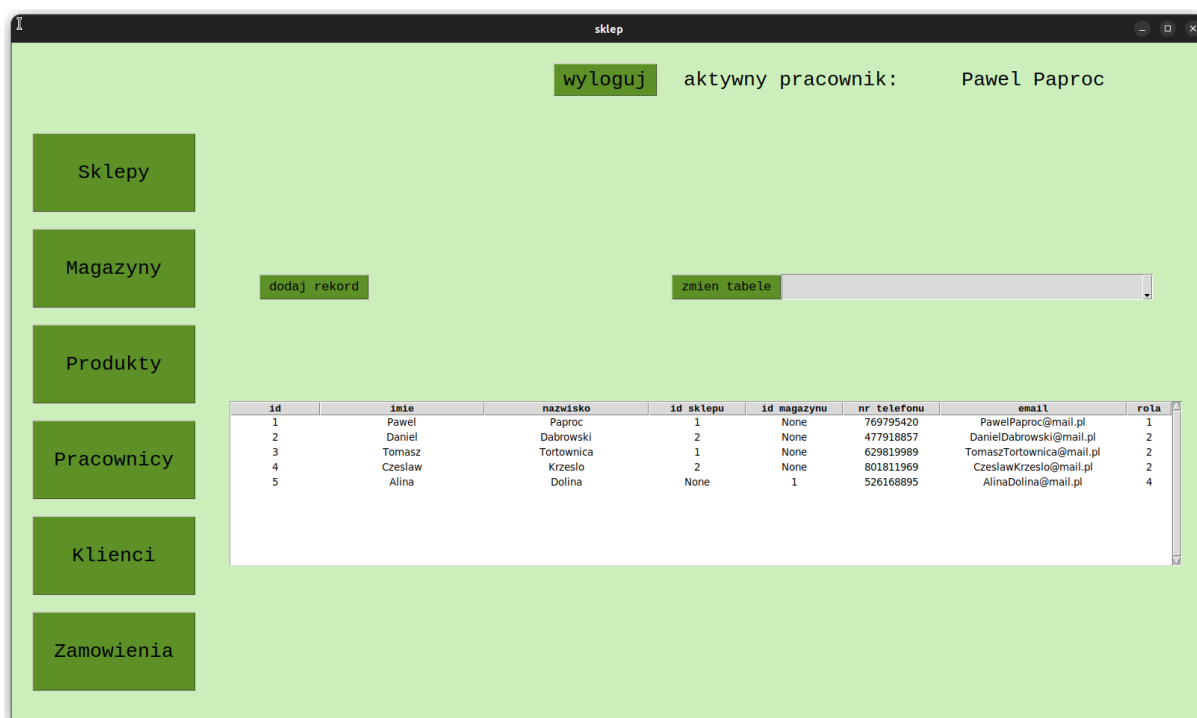
id produktu	nazwa	id sklepu	ilosc	miasto	ulica	nr domu	nr mieszkania
1	wlóczka	50	1	Krakow	Szklana	23	1
1	wlóczka	50	2	Warszawa	Długa	7	2
2	wlóczka	50	1	Krakow	Szklana	23	1
2	wlóczka	50	2	Warszawa	Długa	7	2
3	wlóczka	50	1	Krakow	Szklana	23	1
3	wlóczka	50	2	Warszawa	Długa	7	2
4	wlóczka	50	1	Krakow	Szklana	23	1
4	wlóczka	50	2	Warszawa	Długa	7	2
5	wlóczka	50	2	Warszawa	Długa	7	2
5	wlóczka	50	1	Krakow	Szklana	23	1

Rysunek 15: Raport oparty na widoku, pokazuje produkty w sklepach.

Po kliknięciu przycisku 'Pracownicy' widzimy interfejs [17](#).



Rysunek 16: Raport oparty na widoku, pokazuje produkty w magazynach.



Rysunek 17: Zakładka pracownicy.

Następnie możemy otworzyć formularz wprowadzający dane do tabeli pracownik (rysunek 18) albo wyświetlić widoki związane z tabelą produkt (rysunki 19 i 20).

formularz

id

imie

nazwisko

id sklepu

id magazynu

nr telefonu

email

rola

Rysunek 18: Formularz dodający pracownika (wprowadzanie danych do tabeli pracownik).

sklep

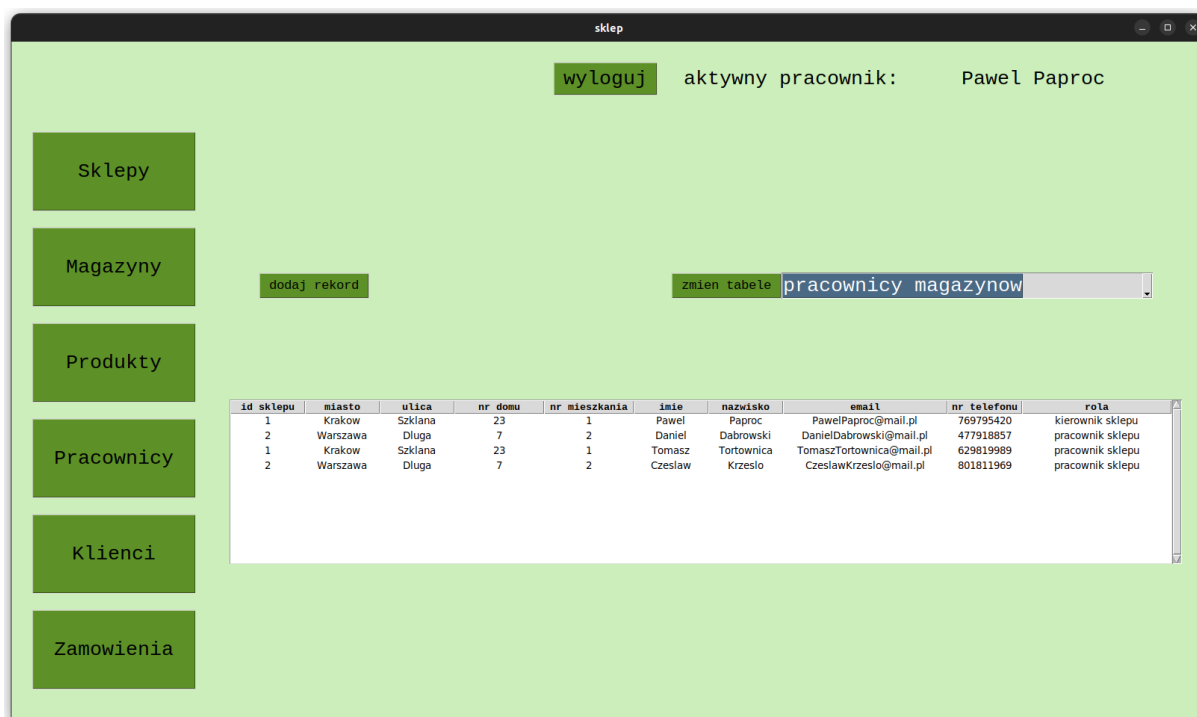
aktywny pracownik: Pawel Paproc

pracownicy sklepow

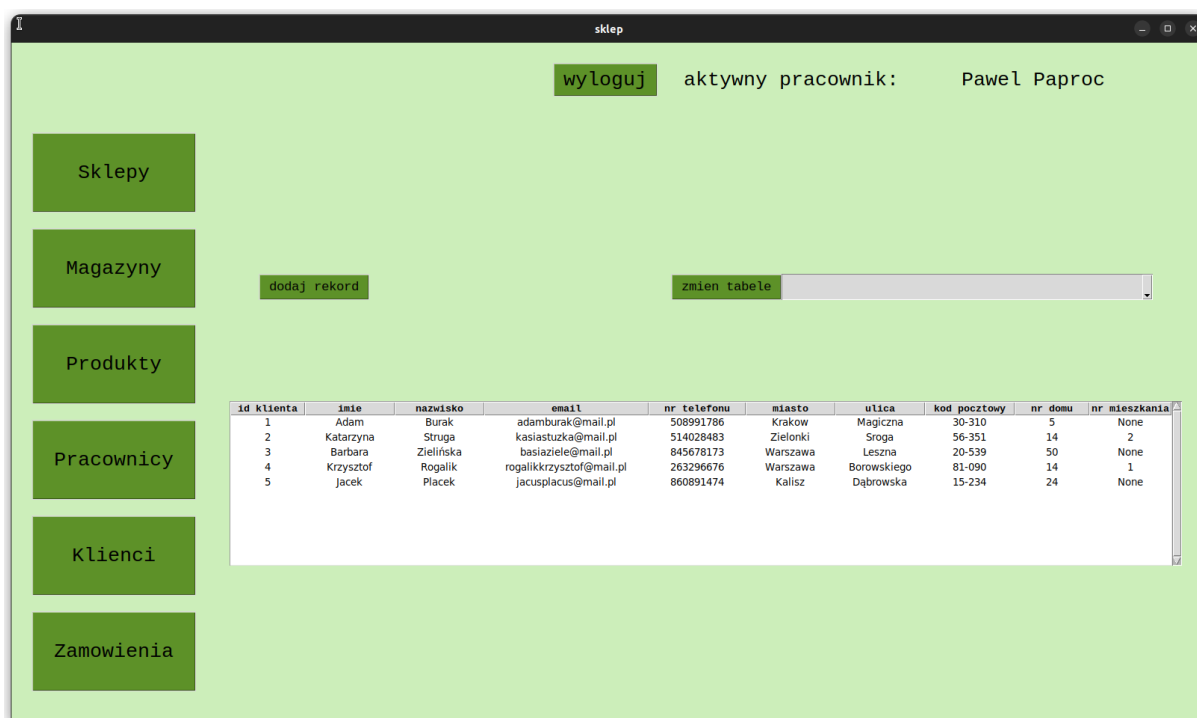
id sklepu	miasto	ulica	nr domu	nr mieszkania	imie	nazwisko	email	nr telefonu	rola
1	Krakow	Szklana	23	1	Pawel	Paproc	PawelPaproc@mail.pl	769795420	kierownik sklepu
2	Warszawa	Długa	7	2	Daniel	Dabrowski	DanielDabrowski@mail.pl	477918857	pracownik sklepu
1	Krakow	Szklana	23	1	Tomasz	Tortownica	TomaszTortownica@mail.pl	629819989	pracownik sklepu
2	Warszawa	Długa	7	2	Czesław	Krzesło	CzeslawKrzeslo@mail.pl	801811969	pracownik sklepu

Rysunek 19: Raport oparty na widoku, pokazuje pracowników sklepów.

Po kliknięciu przycisku 'Klienci' widzimy interfejs [21](#).



Rysunek 20: Raport oparty na widoku, pokazuje pracowników magazynów.



Rysunek 21: Zakładka klienci.

Następnie możemy otworzyć formularz wprowadzający dane do tabeli klient (rysunek 25) albo wyświetlić widok związany z tabelą klient (rysunki 23).

formularz

id klienta

imie

nazwisko

email

nr telefonu

miasto

ulica

kod pocztowy

nr domu

nr mieszkania

dodaj zamknij

Rysunek 22: Formularz dodający klienta (wprowadzanie danych do tabeli klient).

sklep

wyloguj aktywny pracownik: Pawel Paproc

Sklepy

Magazyny

Produkty

Pracownicy

Klienci

Zamowienia

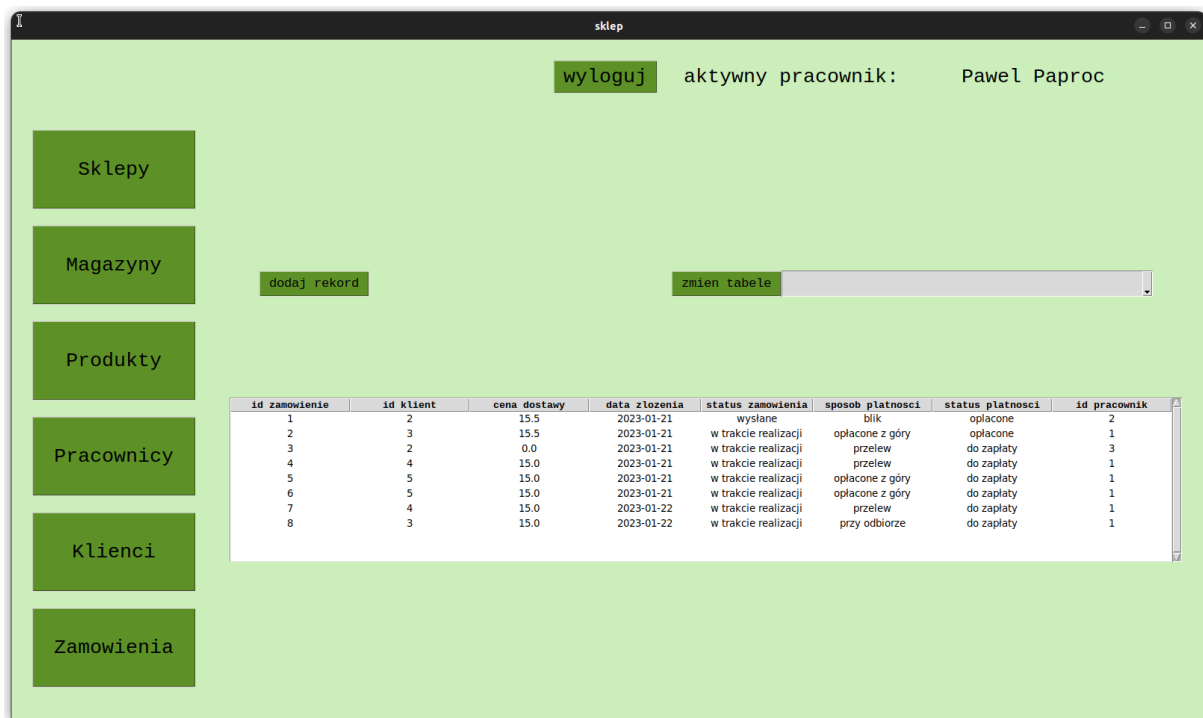
dodaj rekord

zmien tabela zamowienia klientow

id klienta	imie	nazwisko	email	nr telefonu	miasto	ulica	kod pocztowy	nr domu	nr mieszkania
1	Adam	Burak	adamburak@mail.pl	508991786	Krakow	Magiczna	30-310	5	None
2	Katarzyna	Struga	kasiastuzka@mail.pl	514028483	Zielonki	Sroga	56-351	14	2
3	Barbara	Zielińska	basiaziele@mail.pl	845678173	Warszawa	Leszna	20-539	50	None
4	Krzysztof	Rogalik	rogalikkrzysztof@mail.pl	263296676	Warszawa	Borowskiego	81-090	14	1
5	Jacek	Placek	jacuspacus@mail.pl	860891474	Kalisz	Dąbrowska	15-234	24	None

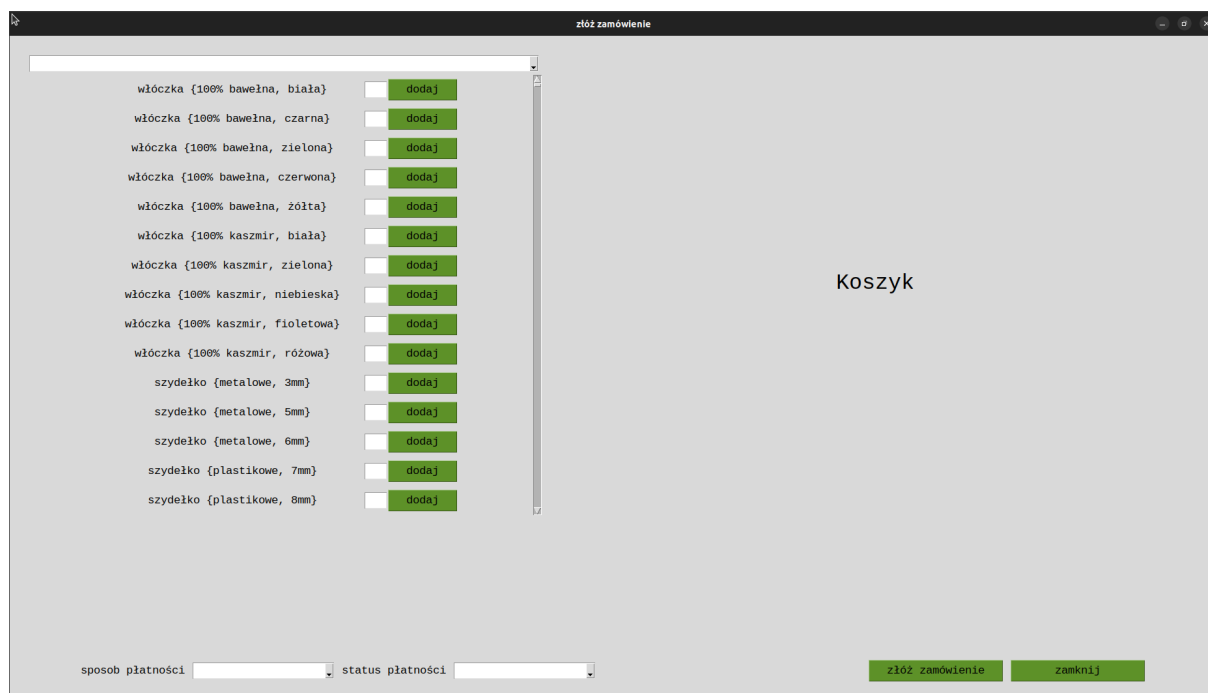
Rysunek 23: Raport oparty na widoku, pokazuje zarys zamówień klientów.

Po kliknięciu przycisku 'Zamówienia' widzimy interfejs 24.

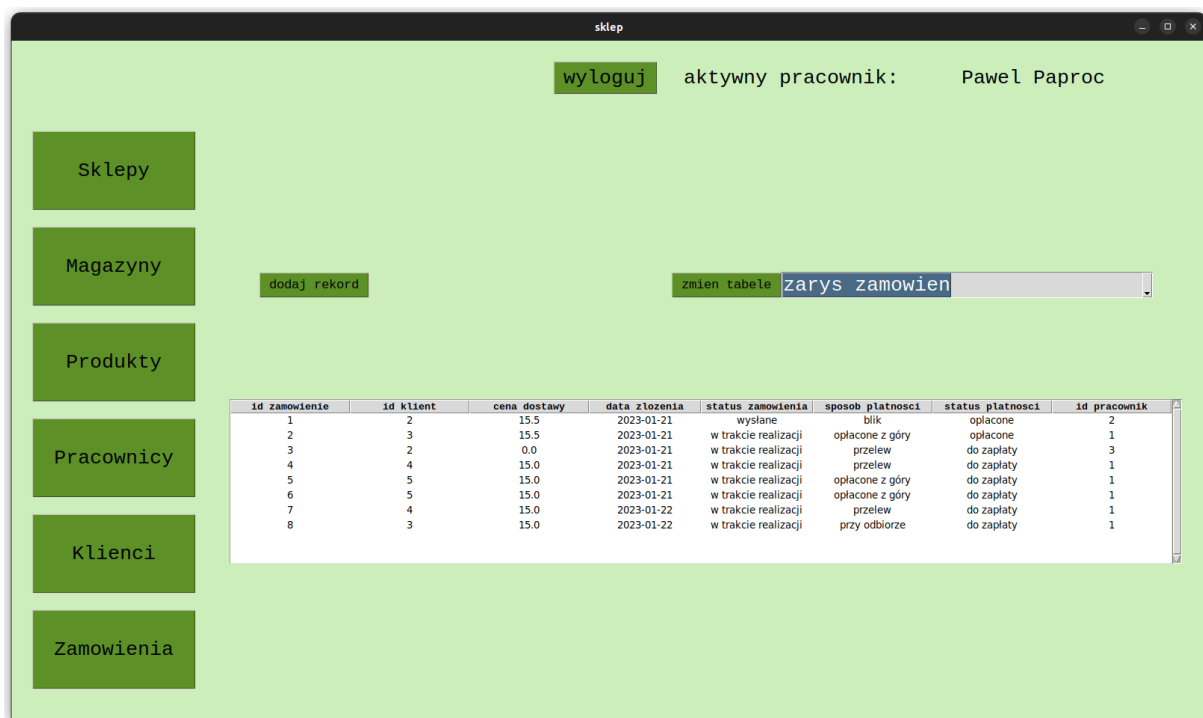


Rysunek 24: Zakładka zamówienia.

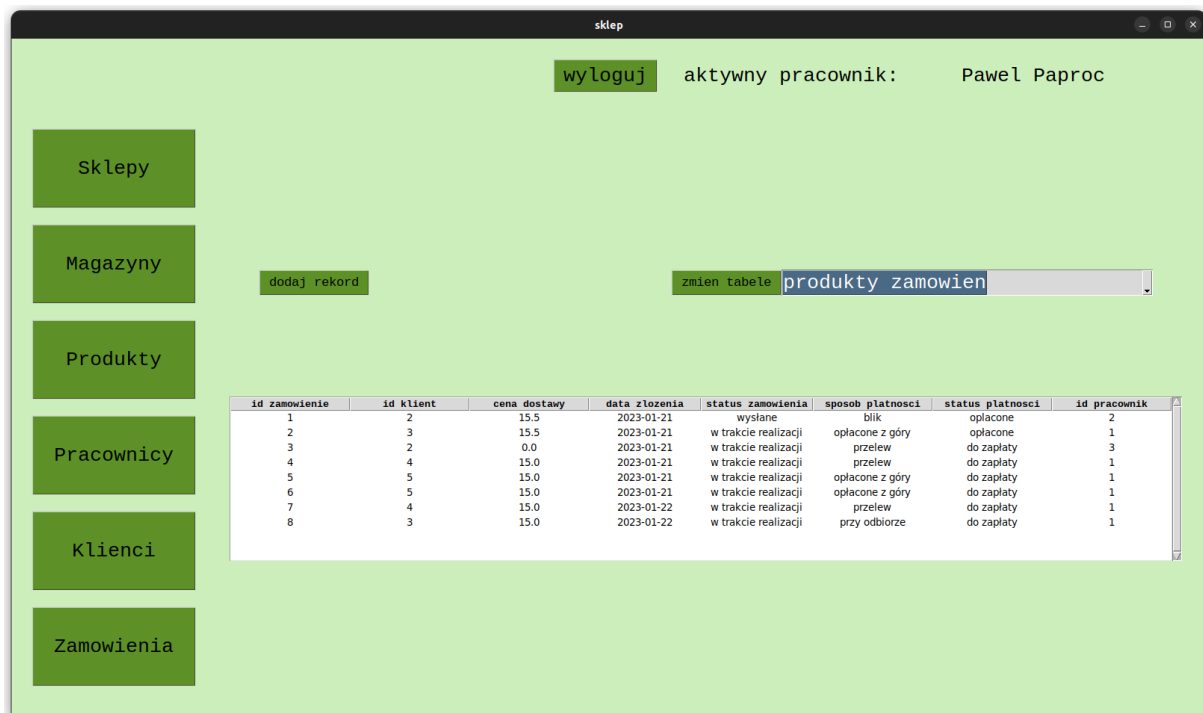
Następnie możemy otworzyć formularz wprowadzający dane do tabeli zamówienia (rysunek ??) albo wyświetlić widoki związane z tabelą zamówienia (rysunki 26, 27 i 28).



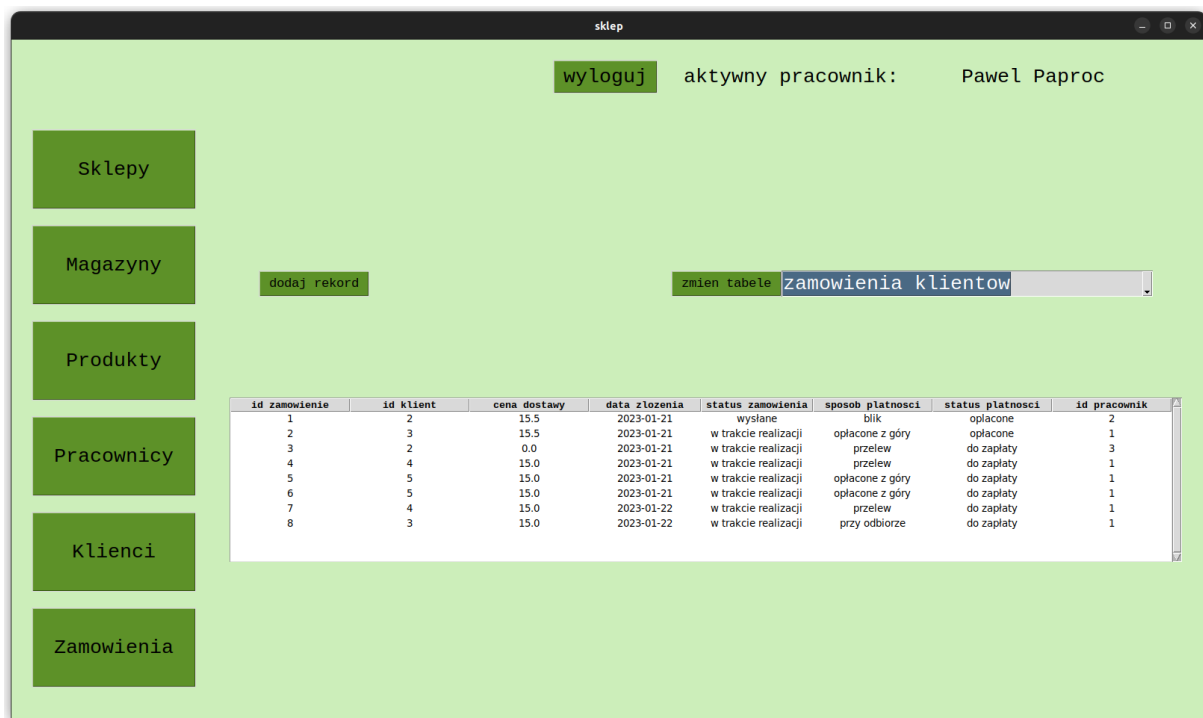
Rysunek 25: Formularz dodający zamówienie (wprowadzanie danych do tabeli zamówienie).



Rysunek 26: Raport oparty na widoku, pokazuje zarys zamówień.



Rysunek 27: Raport oparty na widoku, pokazuje produkty zamówień.



Rysunek 28: Raport oparty na widoku, pokazuje zarys zamówień.

5 Dokumentacja

5.1 Wprowadzanie danych

Wprowadzanie danych odbywało się ręcznie. Do bazy danych dodano kilka sklepów i jeden magazyn wraz z ich inwentarzem. Wprowadzono również dane osobowe pracowników i klientów, oraz zamówienia. W folderze sql znajdują się pliki potrzebne do stworzenia i uzupełnienia, jak również zapewnienia funkcjonalności bazy.

- check_data.sql - plik zawierający polecenia CHECK, odpowiadające za walidację danych,
- create_database.sql - plik zawiera polecenia tworzące wszystkie tabele w bazie danych,
- foreign_keys.sql - w pliku znajdują się polecenia ustawiające klucze obce, uspoźniające bazę,
- functions.sql - funkcje pomocnicze tworzące adres email i losowy numer telefonu,
- insert_data.sql - plik zawiera polecenia INSERT uzupełniające bazę danymi,
- triggers.sql - trigger odpowiadający za wyciągnięcie odpowiednich przedmiotów z magazynu,
- views.sql - w pliku znajdują się widoki potrzebne do utworzenia raportów.

Aplikacja umożliwia pracownikowi na wprowadzenie danych do bazy za pomocą formularzy.

5.2 Opracowanie dokumentacji technicznej

5.2.1 Uruchomienie programu

Do uruchomienia programu potrzebne są następujące paczki:

- tk,
- pycogp2.

z pliku *requirements.txt*, które instaluje się komendą:

```
pip3 install -r requirements.txt
```

Potrzebne pliki znajdują się w repozytorium git pod linkiem <https://github.com/przetoka/BD1-Project.git>.

Po zainstalowaniu odpowiednich modułów i pobraniu plików z repozytorium program uruchamia się poprzez wywołanie komendy:

```
python3 main.py
```

5.2.2 Widoki

- `pracownicy_sklepow` - wyświetla dane sklepu i pracownika,
- `pracownicy_magazynow` - wyświetla dane sklepu i pracownika,
- `produkty_w_sklepach` - wyświetla dane produktu i w jakim sklepie się znajduje oraz w jakiej ilości,
- `produkty_w_magazynach` - wyświetla dane produktu i w jakim sklepie się znajduje oraz w jakiej ilości,
- `zamowienia_zarys` - wyświetla dane klienta i ogólne dane o jego poszczególnych zamówieniach,
- `zamowienia_produkty` - wyświetla dane klienta oraz produkty z zamówień, wykorzystuje funkcję agregującą (liczy łączną cenę zamówionych produktów o danym id),
- `zamowienia_klientow` - wyświetla dane klientów, przedmiot, który zamówili najczęściej rany (i w jakiej ilości) oraz łączną ilość zamówień, wykorzystuje funkcje agregujące (liczy łączną cenę zamówionych produktów o danym id, znajduje największą wartość i zlicza łączną ilość zamówień klienta).

5.2.3 Triggery

- `order_products` - sprawdza czy w magazynie jest odpowiednia ilość produktów do zamówienia i odejmuje je z zaopatrzenia.

5.3 Wykaz literatury

Do wykonania projektu wykorzystano następujące źródła:

- wykłady dr. inż. Antoniego Dydejczyka,
- materiały przygotowujące do laboratoriów mgr. inż. Andrzeja Lemańskiego,
- <https://docs.python.org/3/library/tk.html> - dokumentacja tkinter'a,
- <https://docs.python.org/3/> - dokumentacja Python'a