

eRecepty

Rafał Gubernat

1 Sformułowanie zadania projektowego

Przedmiotem projektu jest realizacja systemu wspomagającego wypisywanie recept. Zostanie on zrealizowany jako aplikacja internetowa. Po wyszukaniu (lub dodaniu nowego) pacjenta lekarz będzie miał możliwość szybkiego (i czytelnego) uzupełnienia blankietu recepty, po czym będzie mógł go wyeksportować do formatu PDF w celu wydruku na dowolnej drukarce komputerowej. Podejście wykorzystujące format PDF umożliwi dokładną wizualizację recepty przed jej fizycznym wydrukiem.

System będzie komunikować się z bazą danych, która zapewni aktualną listę leków, ich dawkę, postać farmaceutyczną, rodzaj opakowania oraz cenę. Baza danych będzie stanowiła źródło informacji o pacjentach — oprócz danych personalnych zawierać będzie szczegóły dotyczące uprzednio przepisanych środków farmaceutycznych. Pomoże to stwierdzić lekarzowi aktualnie przepisywane przez niego lek nie koliduje z lekami, które pacjent przyjmował w przeszłości.

2 Analiza stanu wyjściowego

Nie jest mi znane podobne rozwiązanie funkcjonujące w polskiej służbie zdrowia dostarczające pacjentowi wyżej wymienioną funkcjonalność — w większości placówek medycznych lekarze wypisują recepty ręcznie. W Polsce, w najbliższych latach ma zostać wprowadzony podobny system, lecz nie będzie się on opierał na tradycyjnym wydruku, ale na bezpośrednim transferze danych z recepty w obrębie organów służby zdrowia. Analizując historyczne projekty informatyczne wprowadzane w administracji publicznej można domniemywać, iż tak rozległy system nie zostanie szybko wprowadzony, a nawet jeśli dojdzie do jego wdrożenia to przez pewien okres przejściowy będzie funkcjonował równoległe z tradycyjną metodą przepisywania leków.

Platforma *eRecepty* nie jest rozwiązaniem ogólnopolskim (w tym sensie, że każda placówka zdrowia dysponuje własnym serwerem aplikacyjnym i bazą danych), dzięki czemu koszty eksploatacji i wdrożenia są w większości przypadków akceptowalne. Aplikacja obsługuje bardzo małą liczbę użytkowników, co eliminuje zakup kosztownego sprzętu — dla przeciętnego ośrodka zdrowia wystarczy najprostszy serwer, a nawet komputer PC. Bardzo biedne placówki mogą wykorzystywać jeden serwer aplikacyjny, np. w obrębie gminy. Jeżeli ośrodek posiada własny serwer nie występuje również problem dostępności do wydajnego łącza internetowego.

3 Analiza wymagań użytkownika

3.1 Wymagania funkcjonalne

System wymaga jednoznacznej identyfikacji użytkowników w celu kontroli zasobów. Każdy lekarz oraz recepcjonista będzie zobowiązany do założenia konta w systemie (tą czynność będzie mógł wykonać jedynie administrator platformy). Praca z aplikacją będzie wymagała zalogowania. Lekarz będzie mógł wyszukać odpowiedniego pacjenta oraz wypełnić formularz opisujący receptę. Użytkownik systemu dysponuje bazą leków refundowanych oraz podpowiedziami (uzyskanymi z danych historycznych) dotyczącymi, np. dawkowania. Następnie (po wcześniejszej walidacji) lekarz będzie mógł wygenerować receptę. System umożliwi również przeglądanie historycznych recept danego pacjenta.

3.2 Wymagania techniczne

W systemie tym zarówno baza danych jak i aplikacja działają po stronie serwera. Jako serwer aplikacyjny wykorzystany zostanie Glassfish (choć można użyć dowolnego wspierającego technologię Java EE 6), a system bazodanowy to PostgreSQL (wymienione narzędzia są darmowe). Komunikację z bazą danych zapewnia odpowiednio skonfigurowany serwer aplikacyjny. Takie podejście nie wymaga od użytkowników specjalistycznego oprogramowania — wystarczy przeglądarka WWW.

Placówka będzie musiała dysponować własnym serwerem (mała jednostka medyczna będzie mogła uruchomić zarówno system bazodanowy jak i serwer aplikacyjny na fizycznie jednej maszynie — chociaż nie jest to zalecane rozwiązanie). Do komputerów pracowników będzie musiała być doprowadzona sieć komputerowa — łącze internetowe nie będzie wymagane.

4 Scenariusze użycia

1. Logowanie do systemu:

- użytkownik wypełnia pola *login* oraz *hasło*
- system weryfikuje poprawność danych
- po pomyślnej weryfikacji użytkownik uzyskuje dostęp do określonych zasobów
- po nieudanej weryfikacji proszony jest o podanie poprawnych danych
- jeżeli użytkownik zapomni danych wymaganych do zalogowania w systemie wymagany będzie kontakt z administratorem

2. Dodanie pacjenta do bazy danych:

- uzupełnienie formularza z danymi osobowymi
- sprawdzenie czy pacjent nie jest już obecny w systemie (nr PESEL)
- walidacja danych wpisanych przez użytkownika
- poprawne dane zostają wpisane do bazy
- niepoprawne wywołują funkcje, które informują użytkownika o popełnionych błędach

3. Wyszukanie pacjenta w bazie:

- wpisanie numeru PESEL
- przeglądanie historycznych recept

4. Sporządzenie recepty:

- wyszukanie pacjenta (po numerze PESEL)
- wypełnienie formularza recepty
- walidacja formularza
- po poprawnej walidacji następuje zapisanie danych z formularza do bazy, a później generacja recepty w formacie PDF
- po nieudanej weryfikacji użytkownik jest proszony o poprawienie błędów

5 Identyfikacja funkcji

System będzie realizował następujące funkcje w zależności od poziomu uprawnień zalogowanej osoby:

1. Administrator:

- dodanie nowego użytkownika
- określenie poziomu uprawnień

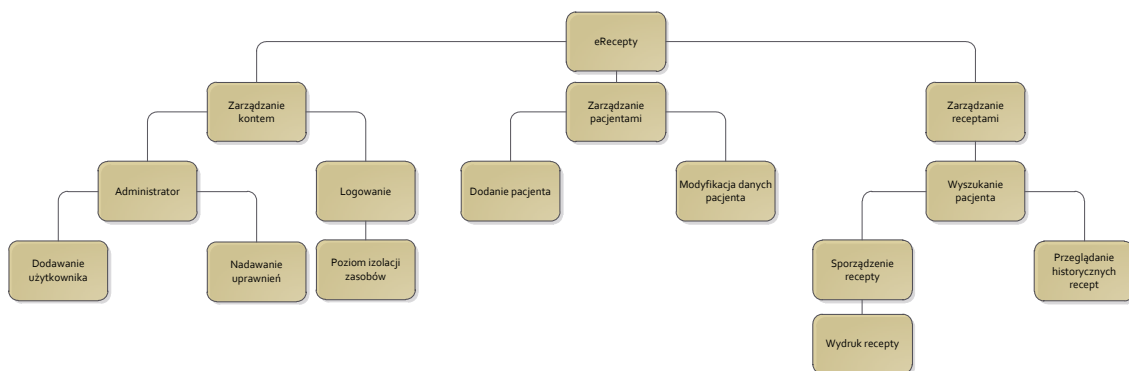
2. Lekarz:

- dodanie pacjenta
- modyfikacja danych pacjenta
- wypisanie recepty
- przejrzanie historycznych recept

3. Recepcjonista:

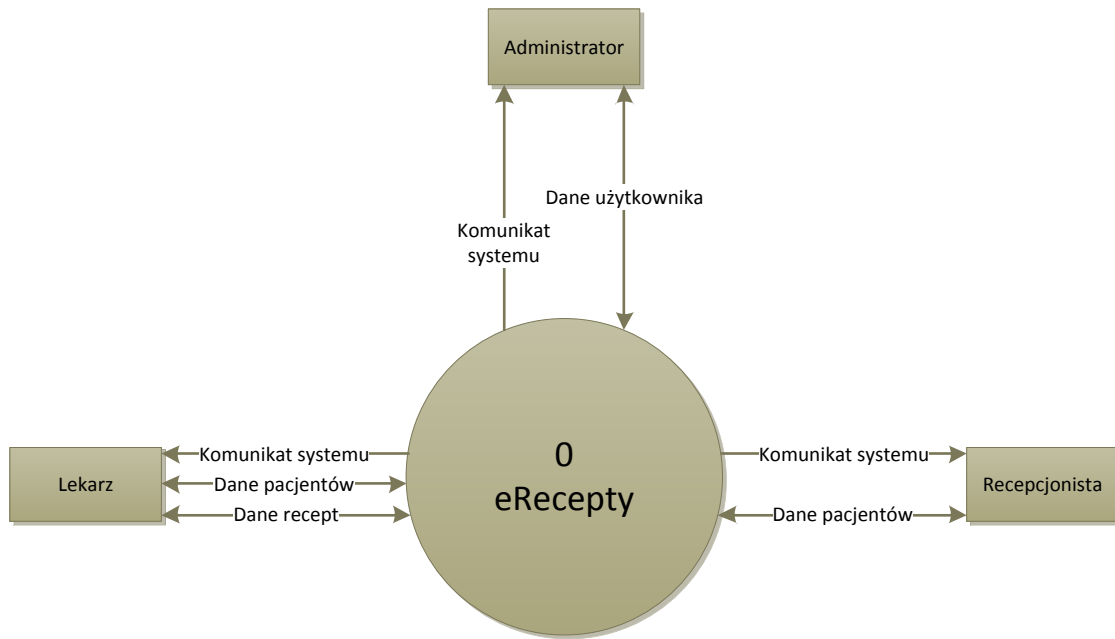
- dodanie pacjenta
- modyfikacja danych pacjenta

6 FHD — diagram hierarchii funkcji

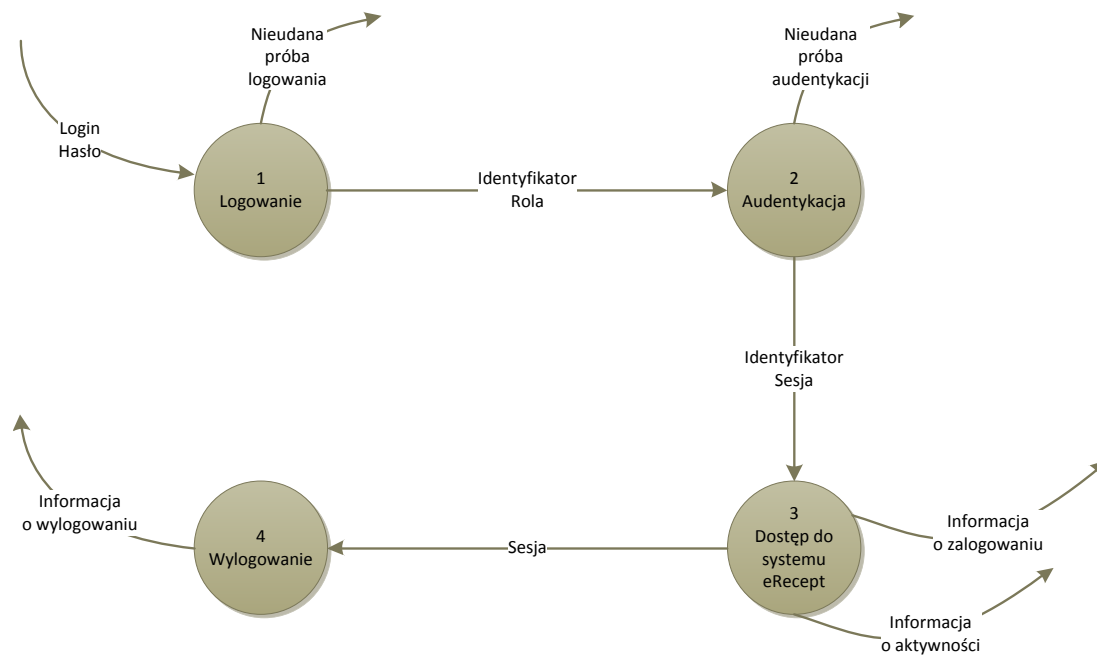


Rysunek 1: Diagram FHD

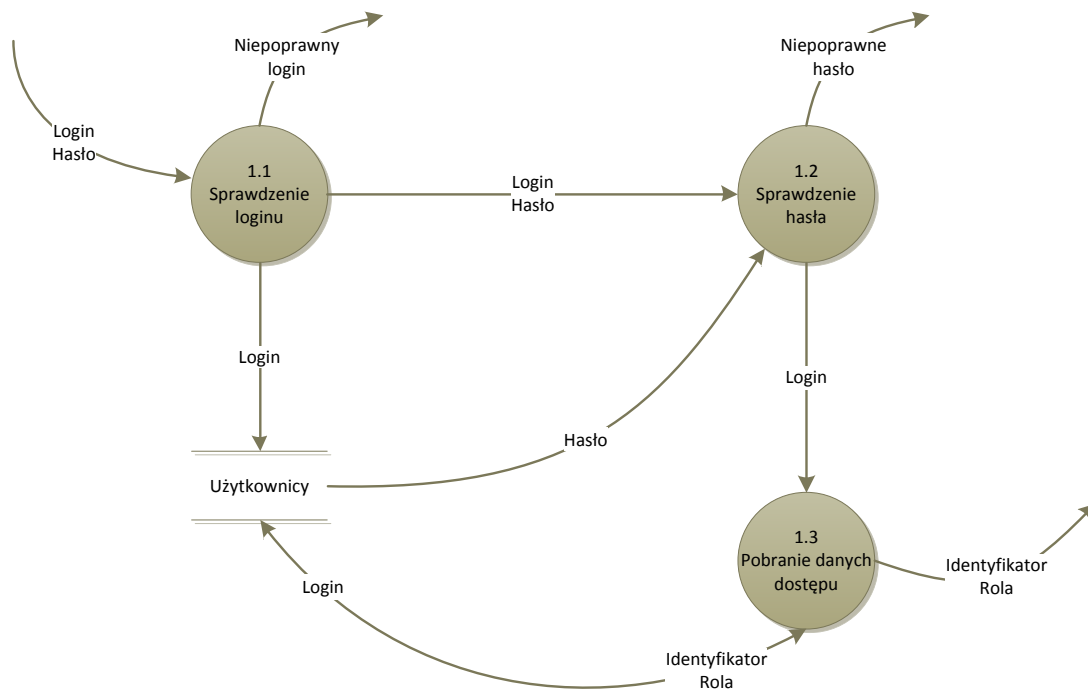
7 DFD — diagramy przepływu danych



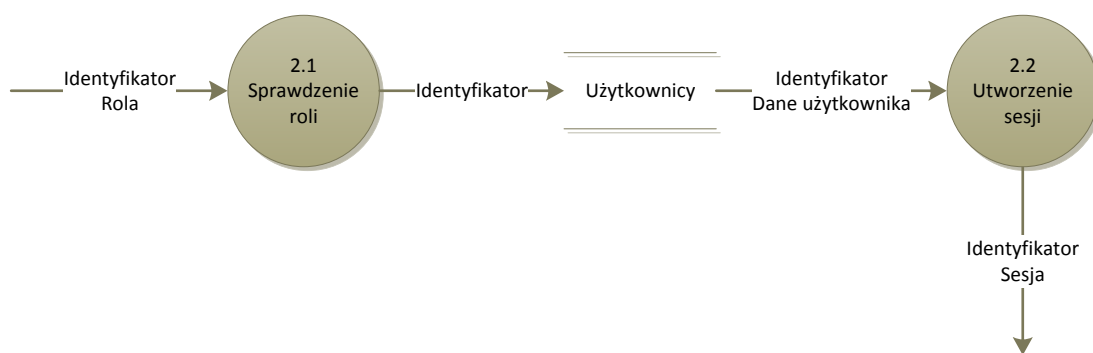
Rysunek 2: Diagram kontekstowy



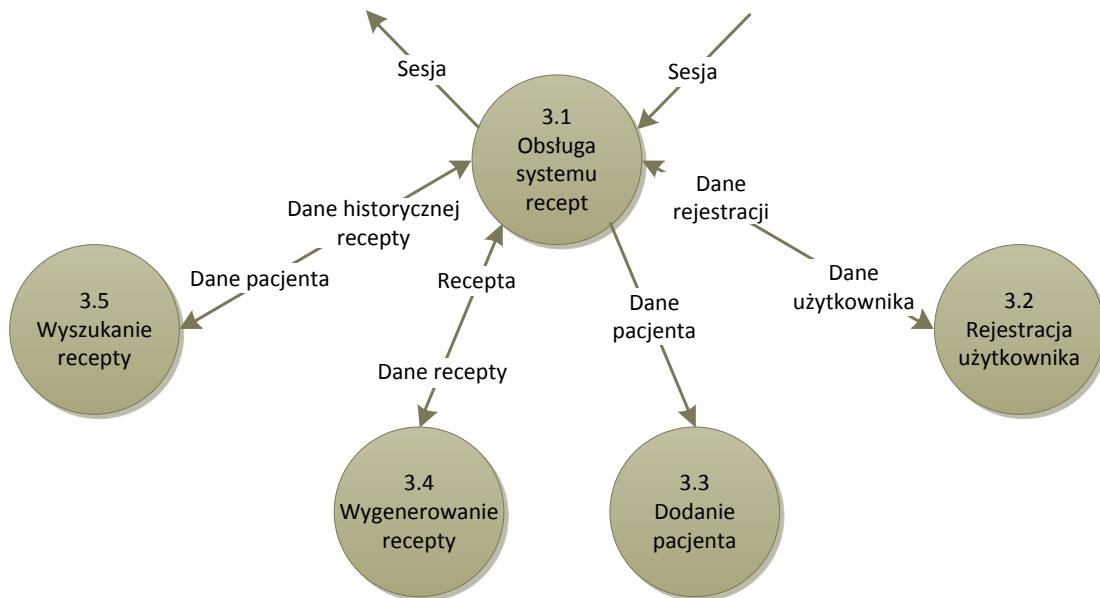
Rysunek 3: Diagram 0



Rysunek 4: Diagram 1



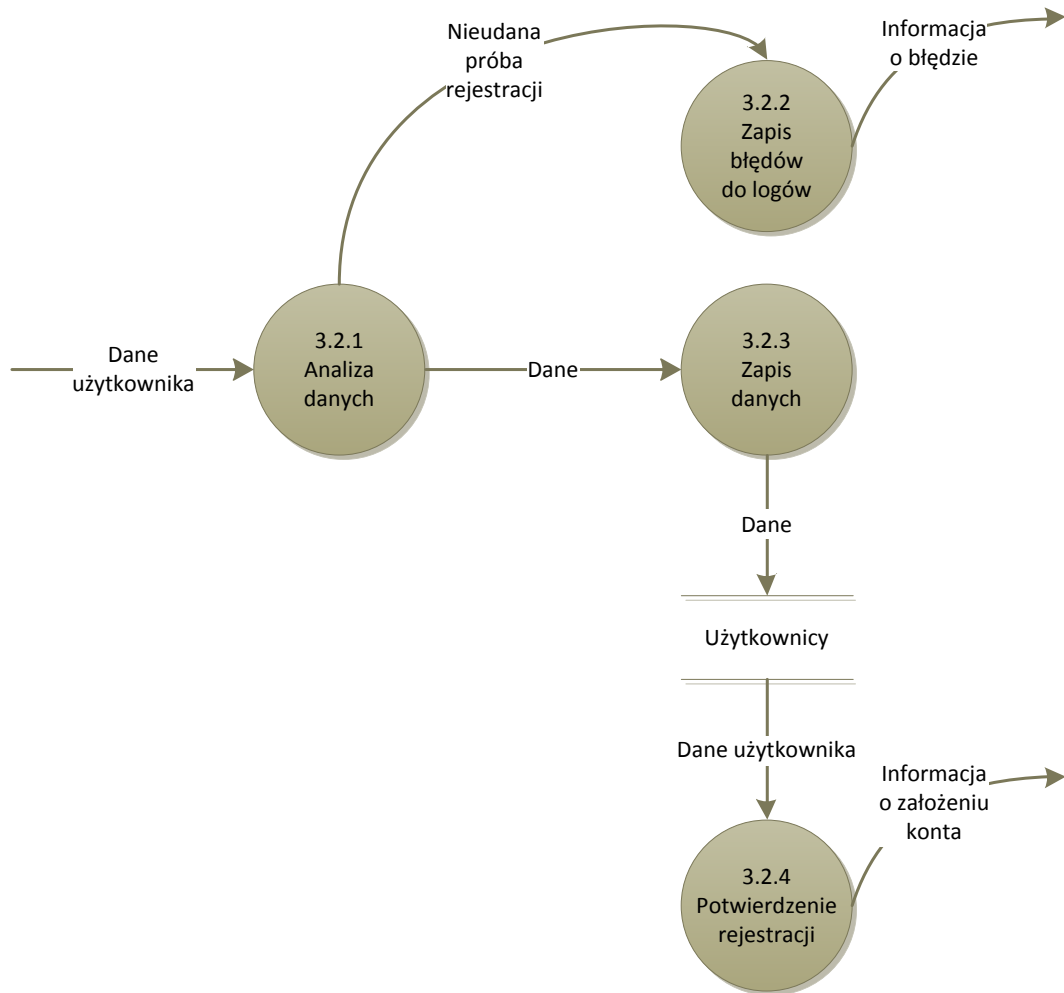
Rysunek 5: Diagram 2



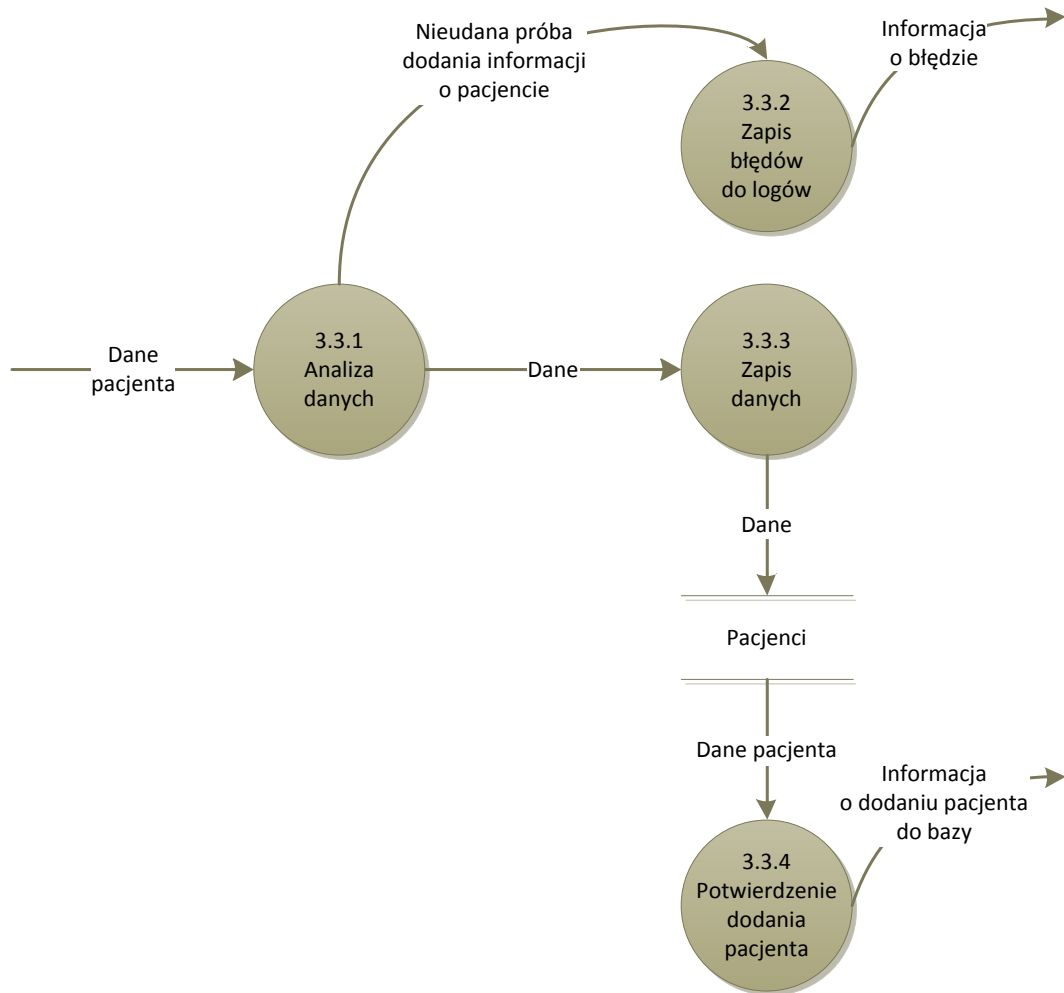
Rysunek 6: Diagram 3



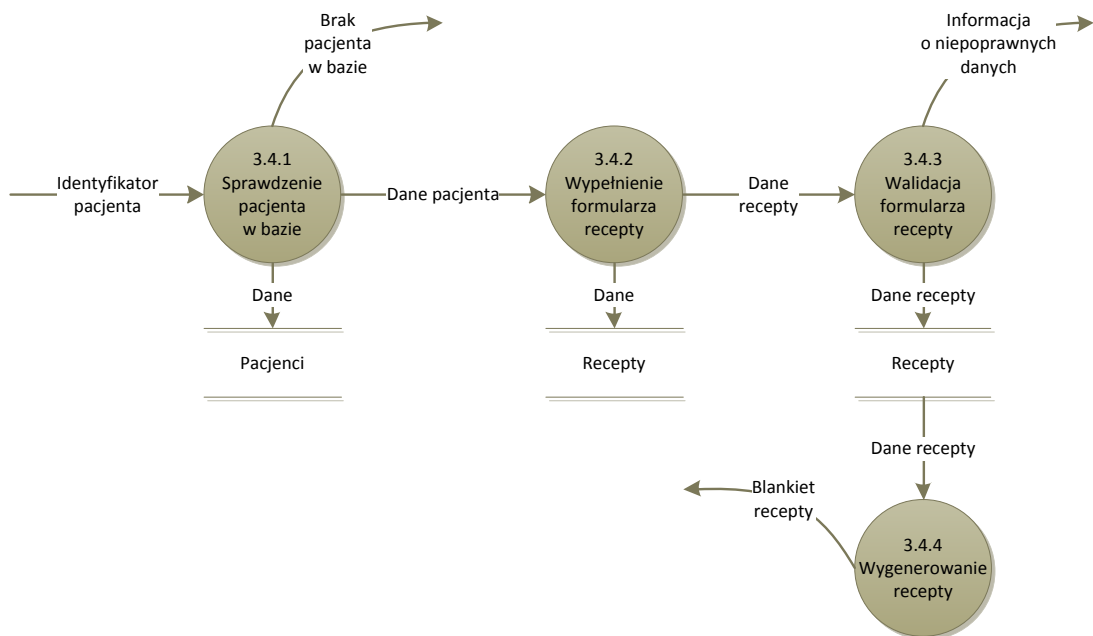
Rysunek 7: Diagram 4



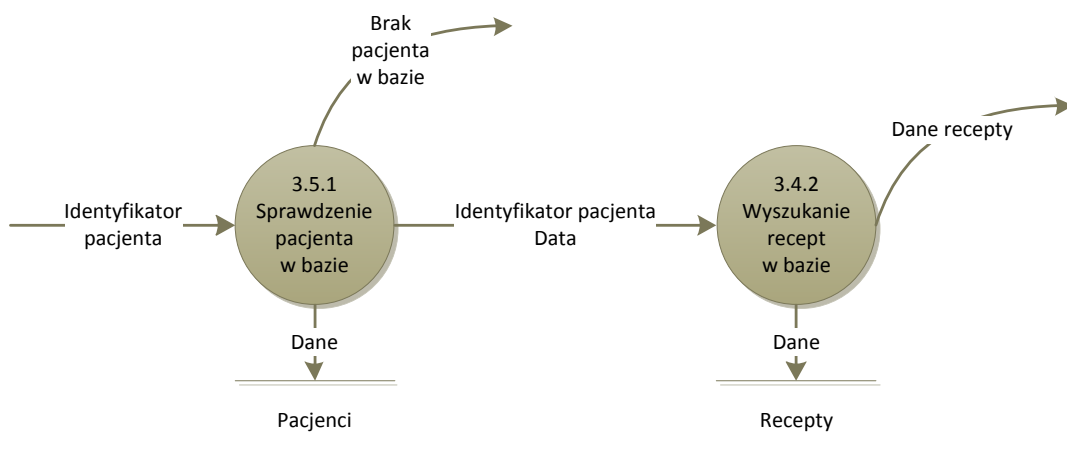
Rysunek 8: Diagram 5



Rysunek 9: Diagram 6



Rysunek 10: Diagram 7

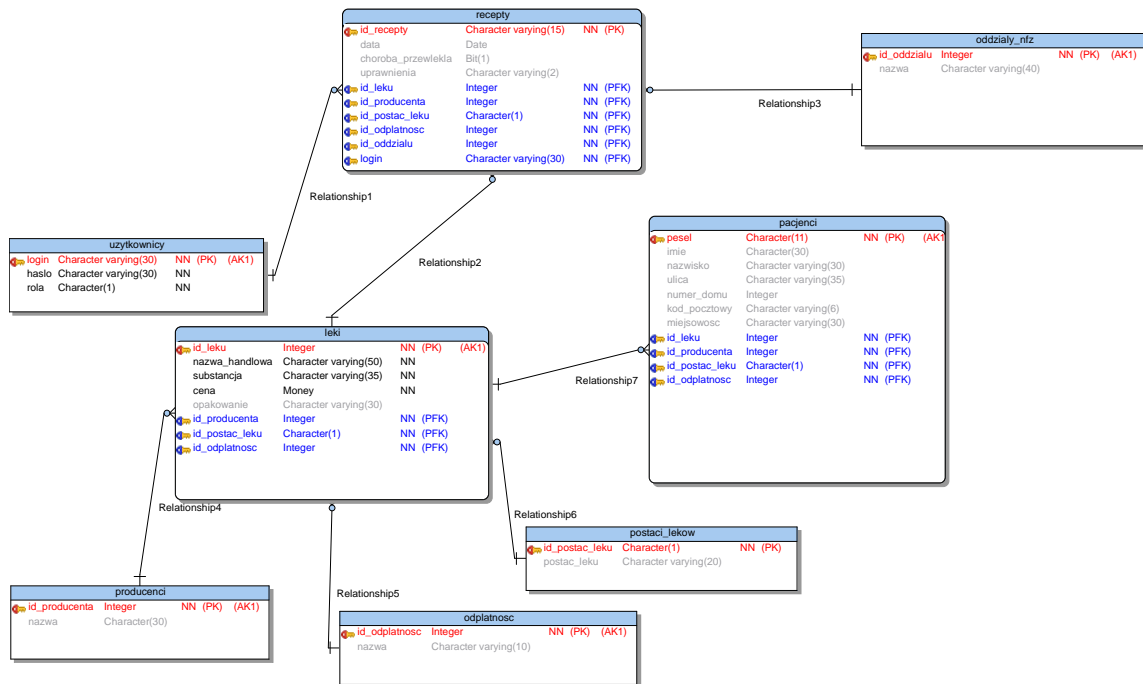


Rysunek 11: Diagram 8

8 Encje i atrybuty

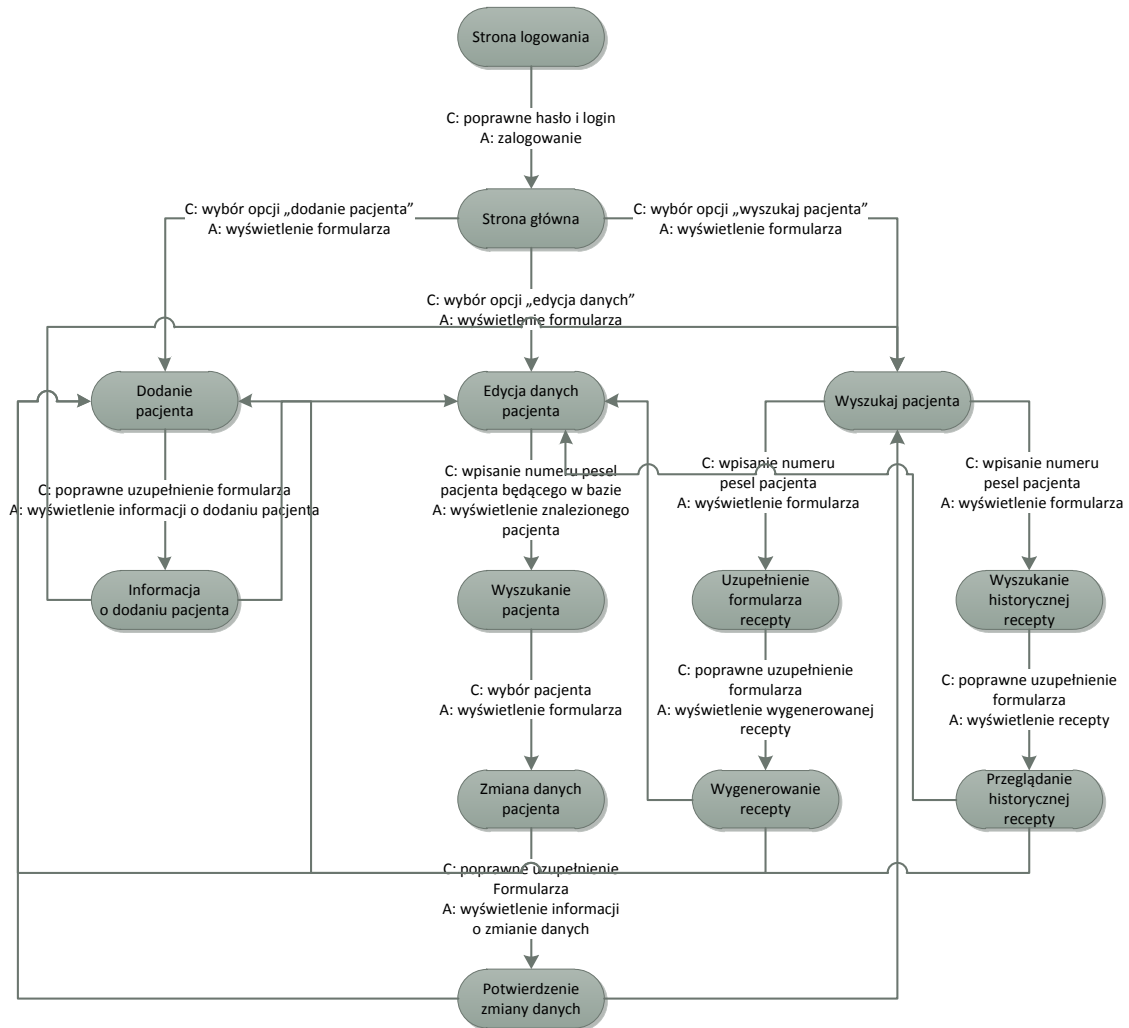
- Użytkownik
- Recepta
- Lek
- Pacjent

9 ERD — diagramy związków encji



Rysunek 12: Diagram 9

10 STD — diagramy przejść pomiędzy stanami



Rysunek 13: Diagram 10