29.05.2019

**Dokumentacja do projektu „Kino Iluzjon”**

Opracował Łukasz Dyduch (s15599) grupa 16c

Spis treści

[1. Wymagania użytkownika 3](#_Toc10132927)

[2. Diagram przypadków użycia 4](#_Toc10132928)

[3. Diagram klas - analityczny 5](#_Toc10132929)

[4. Diagram klas – projektowy 6](#_Toc10132930)

[5. Scenariusz przypadku użycia 7](#_Toc10132931)

[6. Diagram aktywności dla przypadku użycia 7](#_Toc10132932)

[7. Diagram stanu 8](#_Toc10132933)

[8. Diagram interakcji 9](#_Toc10132934)

[9. Projekt GUI 10](#_Toc10132935)

[10. Omówienie decyzji projektowych i skutków analizy dynamicznej 11](#_Toc10132936)

# Wymagania użytkownika

1.1 W systemie należy przechowywać informacje o pracownikach kina i osobach odwiedzających kino. Podział osób jest kompletny.

1.2 Dla każdej z osób trzeba przechowywać jej imię (co najmniej jedno), nazwisko, adres email, numer telefonu i adres zamieszkania. Dla każdego pracownika powinien być również przechowywany jego pesel (musi być unikatowy) oraz numer pracownika. Natomiast dla każdego widza należy dodatkowo przechowywać jego datę urodzenia.

1.3 Bilety sprzedawane są w kasie. System powinien przechowywać informację, który z pracowników sprzedał dany bilet, jakiego rodzaju jest bilet (dostępne to normalny, studencki, szkolny, seniora) i numer biletu (musi być unikatowy).

1.4 Bilety dzielą się na bilety okresowe (karnety) i bilety jednorazowe, podział jest kompletny. Dla okresowych należy przechowywać daty od kiedy i do kiedy ważny jest bilet. Bilety okresowy można również zakupić po wcześniejszym zalogowaniu się, w automacie stojącym w kinie. Konto do logowania zakładane jest w kasie przez obsługę kina.

1.5 Bilety dzielą się również ze względu na ich rodzaj. Wyróżniamy bilety standardowe i bilety VIP. Bilety VIP pozwalają na bezpłatne pobieranie przekąsek ze sklepu w kinie na 15 minut przed seansem i w czasie jego trwania.

1.6 Dodatkowo przy sprzedaży biletu można przypisać do niego dodatkową zniżkę. Zniżka powinna zawierać opis zniżki (warunki jakie trzeba spełnić aby dostać zniżkę) i procent zniżki o który pomniejszona zostanie cena biletu.

1.7 W systemie mają być przechowywane dokładne daty w jakich zarezerwowane są dane sale na projekcje filmową lub wydarzenie kulturowe. Rezerwacje tych samych sal nie mogą kolidować ze sobą w czasie. Przy rezerwacji sali na projekcję filmową należy wziąć pod uwagę fakt czy dana sala wspiera technologie w jakiej wyświetlany będzie film.

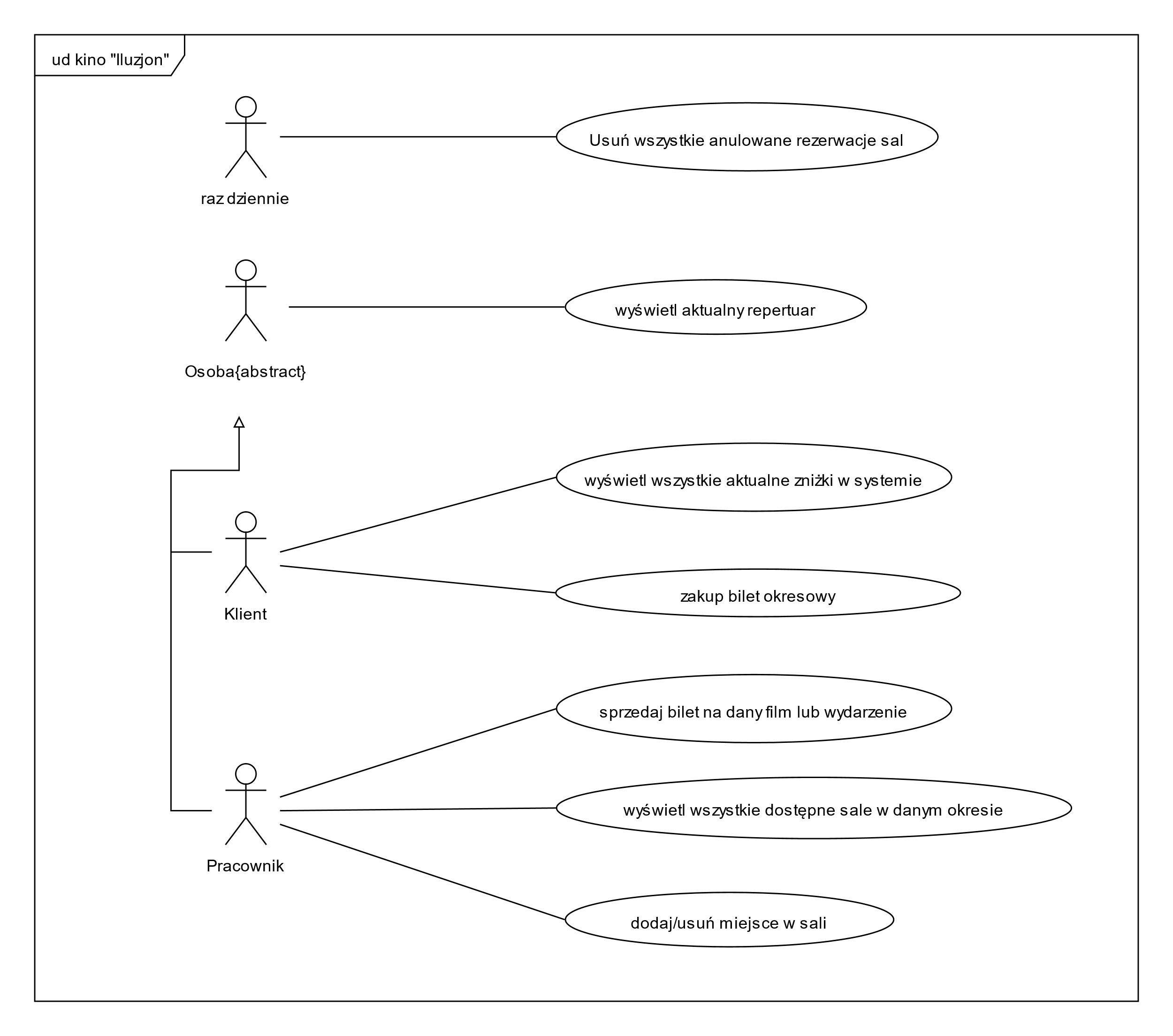
1.8 Sala jest definiowana za pomocą unikatowego numeru sali i jej pojemności. Należy również przechowywać informacje czy dana sala wspiera tylko filmy 2D czy 2D i 3D. W sali znajdują się miejsca, które opisane są poprzez numer miejsca i rząd w którym się znajdują. System powinien zapewnić możliwość dodania lub usunięcia miejsc w razie rozbudowy/zmiany układu sali. W tym czasie sala powinna być widoczna w systemie jako niedostępna.

1.9 System powinien przechowywać informacje o każdym wydarzeniu odbywającym się w kinie. Wydarzenia te dzielą się na projekcje filmowe i wydarzenia kulturowe. Podział jest kompletny. Dla wszystkich wydarzeń można opcjonalnie podać informacje jaka sala jest wymagana (2D lub 3D). Dla projekcji filmowych należy zapamiętać jaki rodzaj tłumaczenia występuje w filmie (dubbing lub napisy) i cenę bazową biletu. Dla wydarzeń kulturowych należy zapamiętać ich nazwę i opis.

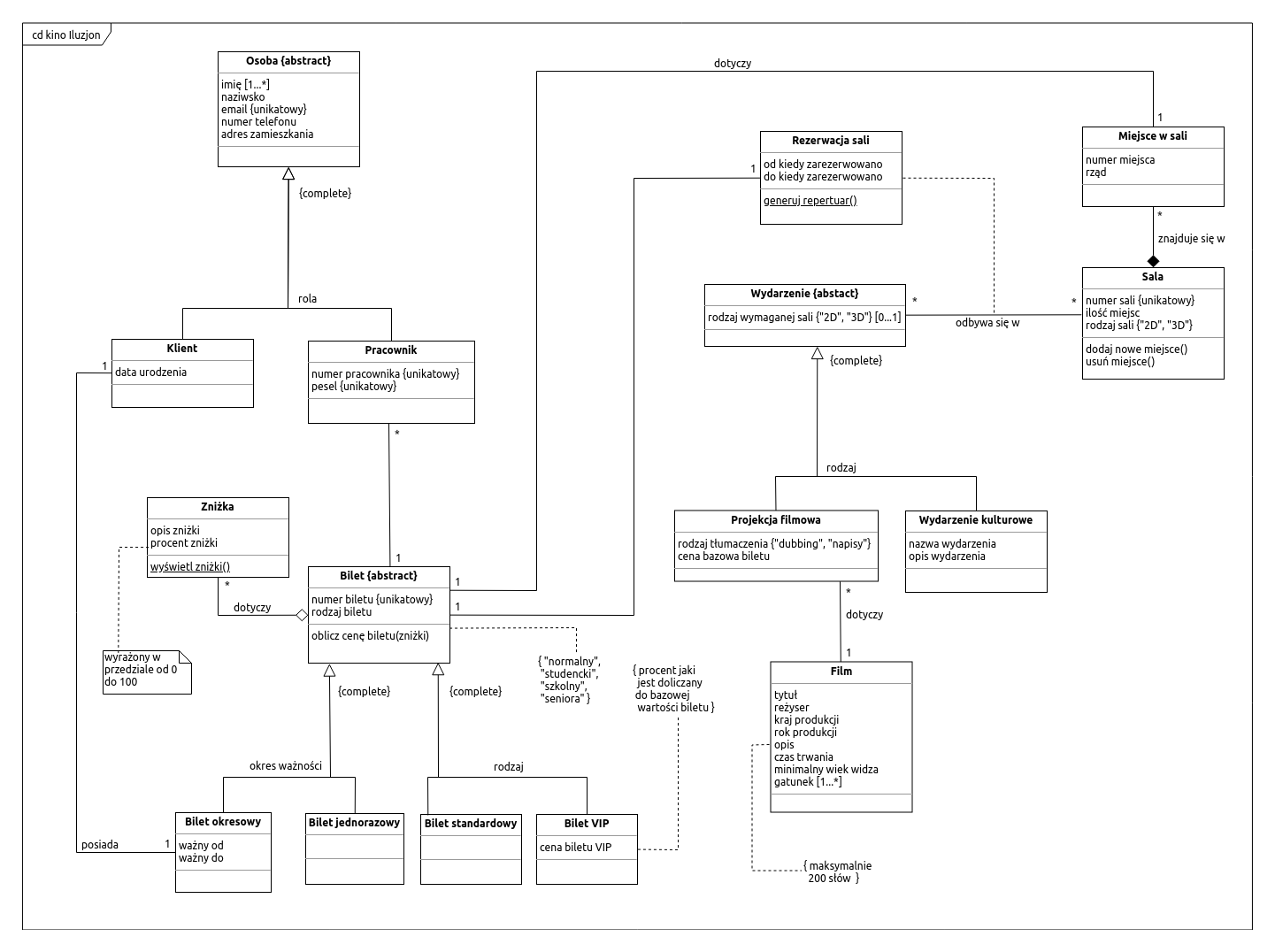
1.10 Pracownik powinien mieć możliwość wygenerowanie aktualnego repertuaru na podstawie aktywnych rezerwacji w systemie.

1.11 Informacje o wyświetlanych filmach należy przechowywać nawet jeśli obecnie nie są one wyświetlane. Każdy film powinien posiadać tytuł, reżysera, kraj produkcji, rok produkcji, opis, czas trwania, minimalny wiek widza oraz przypisane gatunki filmowe.

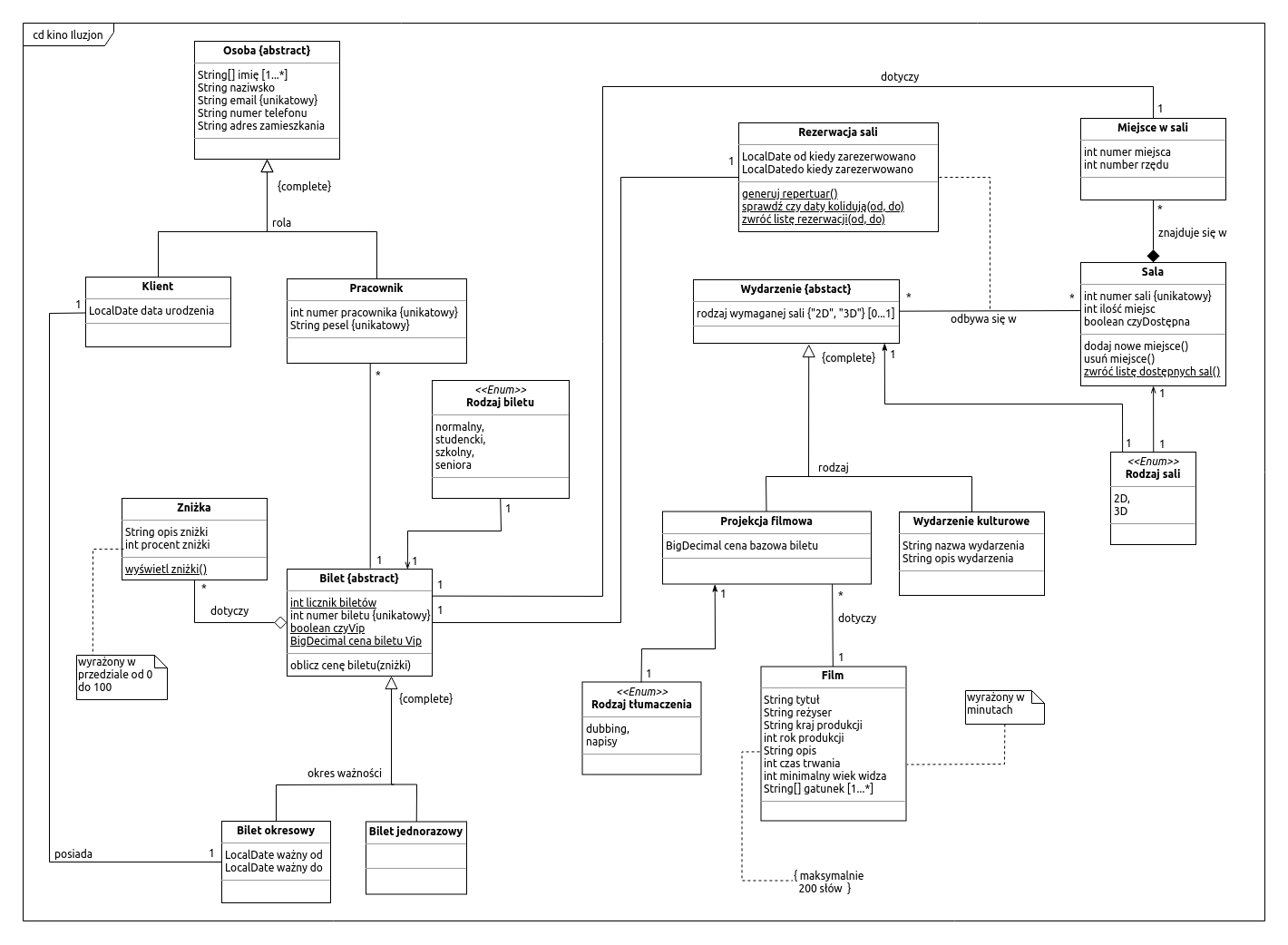
# 2. Diagram przypadków użycia

****

# 3. Diagram klas - analityczny

****

# 4. Diagram klas – projektowy

****

# 5. Scenariusz przypadku użycia

5.1. Przypadek użycia rozpoczyna się, gdy Pracownik kina zechce dodać nowy film i zarezerwować salę na jego projekcję.

5.2. System wyświetla formularz z takimi polami jak tytuł filmu, reżyser, kraj produkcji, opis filmu, czas jego trwania, minimalny wiek widza i gatunek filmu.

5.3. Pracownik wprowadza i zatwierdza dane.

5.4. System sprawdza wprowadzone dane. Jeżeli dane są nieprawidłowe, powtarzany jest krok 5.2 do momentu uzyskania prawidłowych danych lub wybrania opcji anulowania operacji. Po wprowadzeniu prawidłowych danych system wyświetla informacje o pomyślnym dodaniu filmu i pyta czy użytkownik również chciałby dodać rezerwacje sali dla projekcji filmowej nowododanego filmu.

5.5. Pracownik wybiera opcję dodania rezerwacji.

5.6. System wyświetla ekran dodawania rezerwacji, gdzie znajduje się opcja wyboru czy film jest 2D czy 3D, godziny rezerwacji sal w następnym tygodniu, pole z wyborem godziny rozpoczęcia rezerwacji, pole z wyborem godziny zakończenia rezerwacji i lista dostępnych sal, które wspierają wcześniej wybrany format filmu (2D lub 3D).

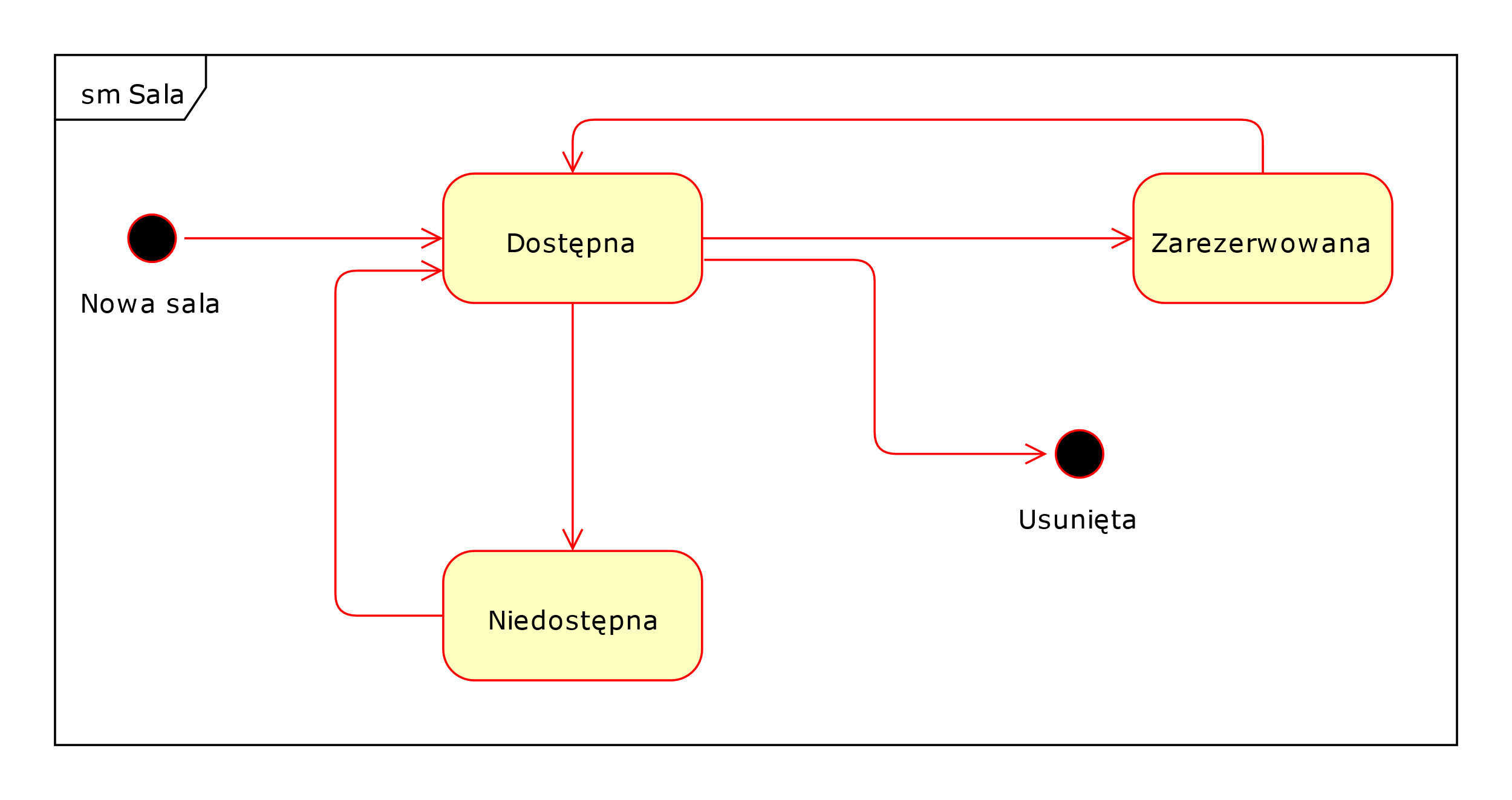
5.7. Pracownik wypełnia formularz i zatwierdza dane.

5.8. System sprawdza wprowadzone dane i wyświetla komunikat o pomyślnym zarezerwowaniu sali i to kończy przypadek użycia.

# Obraz zawierający tekst, mapa Opis wygenerowany automatycznie6. Diagram aktywności dla przypadku użycia

# 

# 7. Diagram stanu

****

# 8. Diagram interakcji

# 9. Projekt GUI

# 10. Omówienie decyzji projektowych i skutków analizy dynamicznej

Atrybuty, których wartości były znane i wyliczalne zostały zastąpione dostępnym w języku programowania Java, typem wyliczeniowym. W taki sposób do diagramu klas zostały dodane klasy takie jak *Rodzaj biletu*, *Rodzaj tłumaczenia* i *Rodzaj sali*.

Dziedziczenie wieloaspektowe z nadklasy Bilet zostało zaimplementowane poprzez *„spłaszczenie”* jednego aspektu do postaci atrybutu *boolean*. Przyjęto takie rozwiązanie z powodu małej ilości danych przechowywanych w klasach tego aspektu.

Analiza dynamiczna realizowanego przypadku użycia poskutkowała pojawieniem się takich metod jak:

- *sprawdź czy daty kolidują(od, do)* w klasie **Rezerwacja Sali**

- *zwróć listę rezerwacji(od, do)* w klasie **Rezerwacja Sali**

- *zwróć listę dostępnych sal()* w klasie **Sala**

Dodatkowo w takie pracy nad projektowym diagramem klas został dodany atrybut *int licznik biletów* mający zapewnić generowanie unikatowych numerów sprzedawanych biletów.