

Multipleks kinowy

Dokumentacja



Wersja 1.3

Grupa habeja, wtorek 17:50, członkowie:

Krzysztof Hardek

Adam Bera

Grzegorz Janosz

Opis projektu

Projekt polega na stworzeniu aplikacji realizującej multipleks kinowy. Aplikacja ma umożliwiać:

- Wprowadzanie osób, w tym z różnymi uprawnieniami,
- Dostęp do aktualnego repertuaru kina,
- System wysyła maile z powiadomieniami
- Statystyki, Wyszukiwanie, Polecanie filmu

Jego realizacja została podzielona na trzy kamienie milowe. Wykonanie każdego z nich skutkuje stworzeniem działającego prototypu

M1

1. Planowany progres:

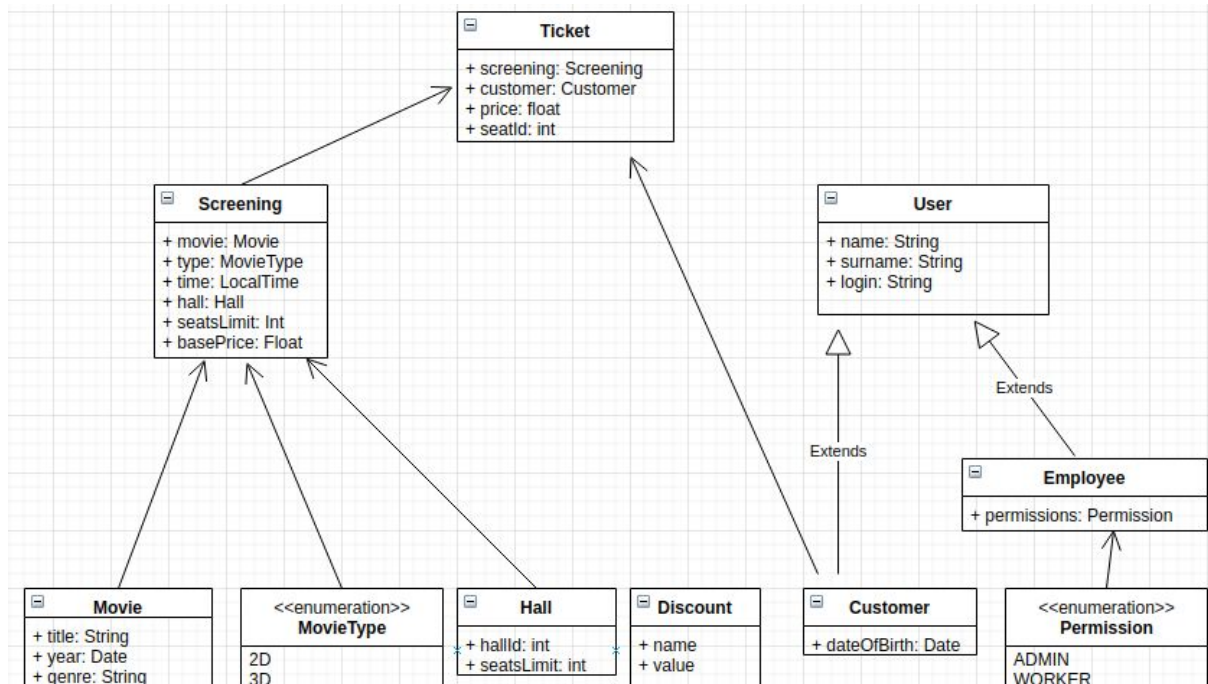
- kompletny model,
- wprowadzanie w aplikacji osób i sal kinowych

2. Podział obowiązków

- Adam Bera - GUI (dla osób i sal)
- Grzegorz Janosz - Baza danych i jej interfejs (dla osób i sal)
- Krzysztof Hardek - Przygotowanie dokumentacji, logika aplikacji (podpięcie bazy i GUI), podstawowe klasy z diagramu (implementacja)
- Cały zespół - Stworzenie modelu (praca koncepcyjna)

3. Co zostało zrobione

Model danych:



Stworzyliśmy model danych w postaci diagramu klas (nie korzystamy z większości dobrodziejstw tego diagramu, służy on jedynie prostej wizualizacji szkieletu naszej aplikacji). Wszystkie atrybuty są póki co publiczne ale we właściwej aplikacji ustawimy takie modyfikatory dostępu jakie będą potrzebne. Pominęliśmy również gettersy, settersy oraz pozostałe metody klas, ponieważ na tym etapie nie jesteśmy w stanie stwierdzić które z nich będą nam potrzebne, a które nie. Nie przedstawiliśmy na diagramie części klas, które znajdują się w naszej aplikacji (np. klasy związane z DAO, GUI). Ze względu na brak metod na diagramie ograniczyliśmy relacje między klasami do asocjacji i dziedziczenia.

Model został w dosyć naturalny sposób narzucony przez specyfikację projektu. Opis klas w nim zawartych wygląda następująco:

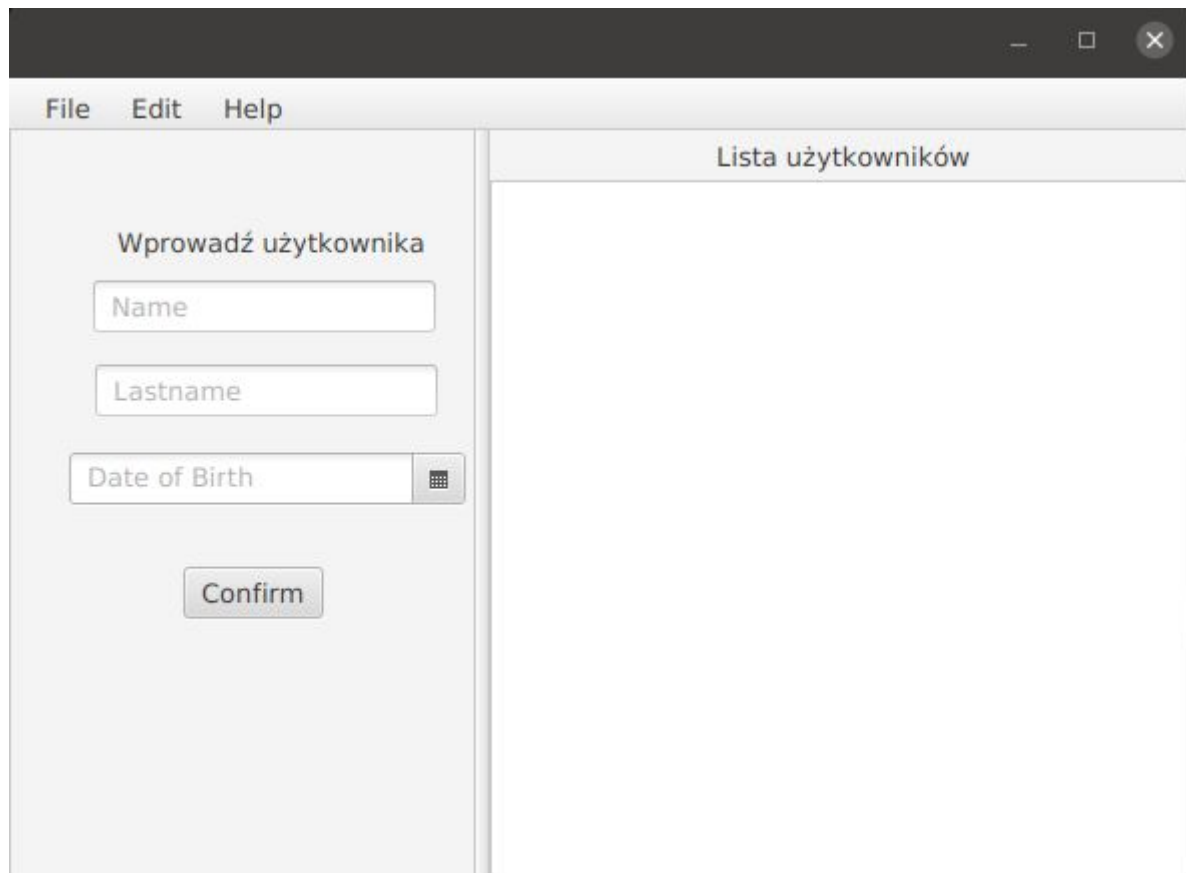
- Movie - Klasa reprezentująca film (rok powstania, tytuł oraz gatunek).
- MovieType - Enum reprezentujący typ filmu. Na chwilę obecną może to być film 2D lub 3D.
- Hall - Sala kinowa. Przechowuje informacje o limicie miejsc.
- Discount - Wszelkiego rodzaju promocje i upusty (wartość może być procentowa lub ilościowa).
- User - Użytkownik systemu.
- Customer - Użytkownik, który jest zwykłym klientem.
- Permissions - Enum związany z uprawnieniami pracowników
- Employee - Pracownik kina.
- Screening - Klasa reprezentująca konkretny seans w konkretnej sali i godzinie. Posiada np. informację i limicie miejsc na dany seans (może on być mniejszy bądź równy od limitu miejsca w sali), informację o cenie podstawowej biletu, która może być regulowana przez zniżki (klasa Discount).
- Ticket - Bilet kupiony (zarezerwowany) na jakieś miejsce na konkretnym seansie przez klienta

Baza danych i jej interfejs:

Korzystamy z firestore - elastycznej nierelacyjnej bazy danej w chmurze (google). Klasa FirestoreDatabase jest używana do komunikacji z bazą i jest ona singletonem. Użyliśmy takiego rozwiązania, ponieważ klient bazy danych powinien być tylko jeden. Klasy HallDao oraz UserDao realizują wzorzec DAO, dzięki któremu możemy swobodnie manipulować bezpośrednią komunikacją z bazą nie zmieniając całej aplikacji.

GUI - dla osób:

Wykonaliśmy GUI tylko dla dodawania użytkowników, ponieważ dla sal będzie to wyglądało analogicznie a nie mamy jeszcze planu jak zorganizować cały interfejs użytkownika



Logika aplikacji:

Zdecydowaliśmy się na zorganizowanie struktury naszej aplikacji zgodnie ze wzorcem MVC, który jest wspierany przez JavaFX.

M2

1. Planowany progres

- Działająca warstwa persystencji
- Autentykacja i autoryzacja
- Zarządzanie salami
- Zarządzenie filmami

2. Podział obowiązków

- Adam Bera - Kontrolery dla zarządzania salami i filmami
- Grzegorz Janosz - Napisanie pozostałych klas DAO, API do autoryzacji i autentykacji
- Krzysztof Hardek - Napisanie pozostałych klas modelu, widoki dla autentykacji i autoryzacji, zarządzania filmami, salami oraz dokumentacja

3. Co zostało zrobione

Model

Dopisane zostały pozostałe klasy z modelu.

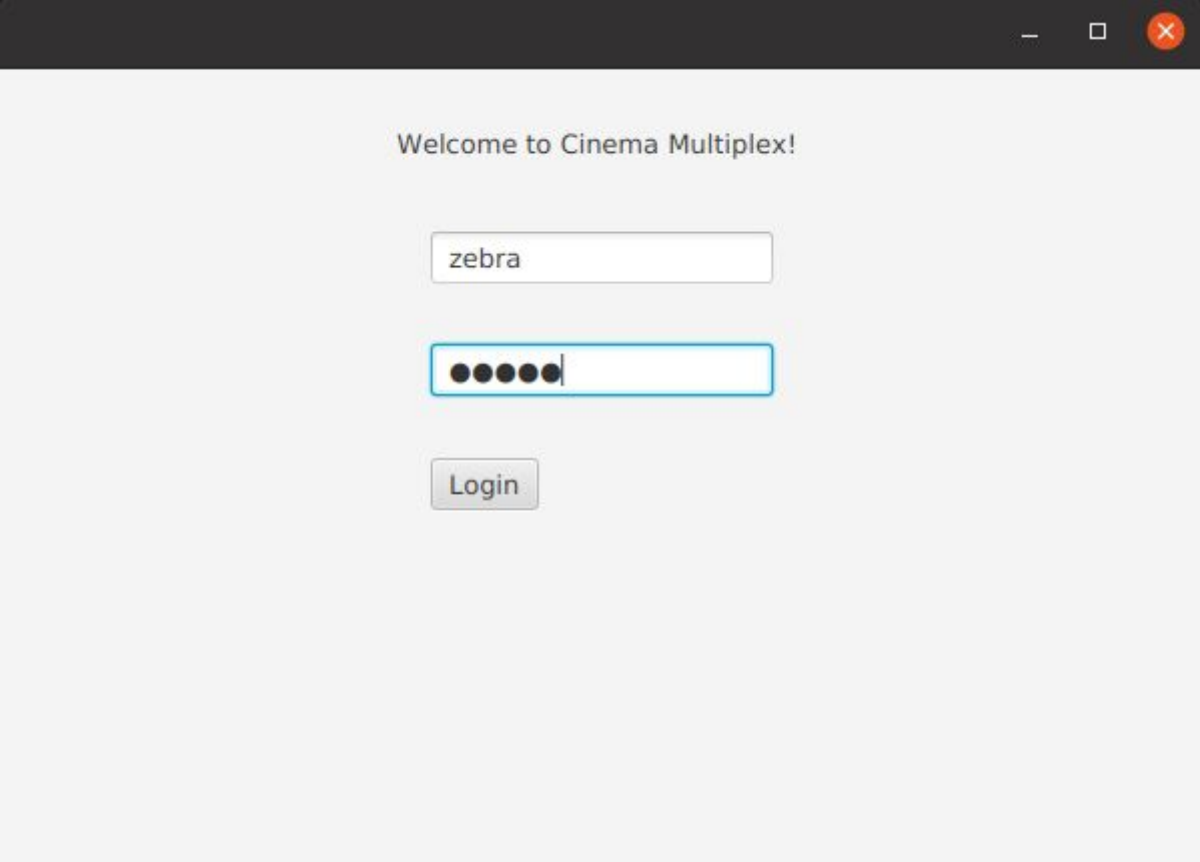
Interfejs danych i API do autoryzacji i autentykacji

Dopisane zostały pozostałe klasy DAO. Wystawiony został interfejs do autentykacji i autoryzacji. W wystawionym API do tworzenia queries wyników z bazy danych zostały użyte tzw. strategie szukające.

Widoki

Dodane zostały widoki dla zarządzania salami i filmami (administrator), seansami (pracownik) oraz logowanie i panele główne dla administratora i pracownika

- Login



The screenshot shows a web application window titled "Welcome to Cinema Multiplex!". It features a login form with two input fields: the first contains the text "zebra", and the second is a password field with five black dots and a cursor. Below the fields is a "Login" button. The window has a dark header bar with standard window controls (minimize, maximize, close) on the right.

- Panel administratora

- *Lista sal*

[illegible]

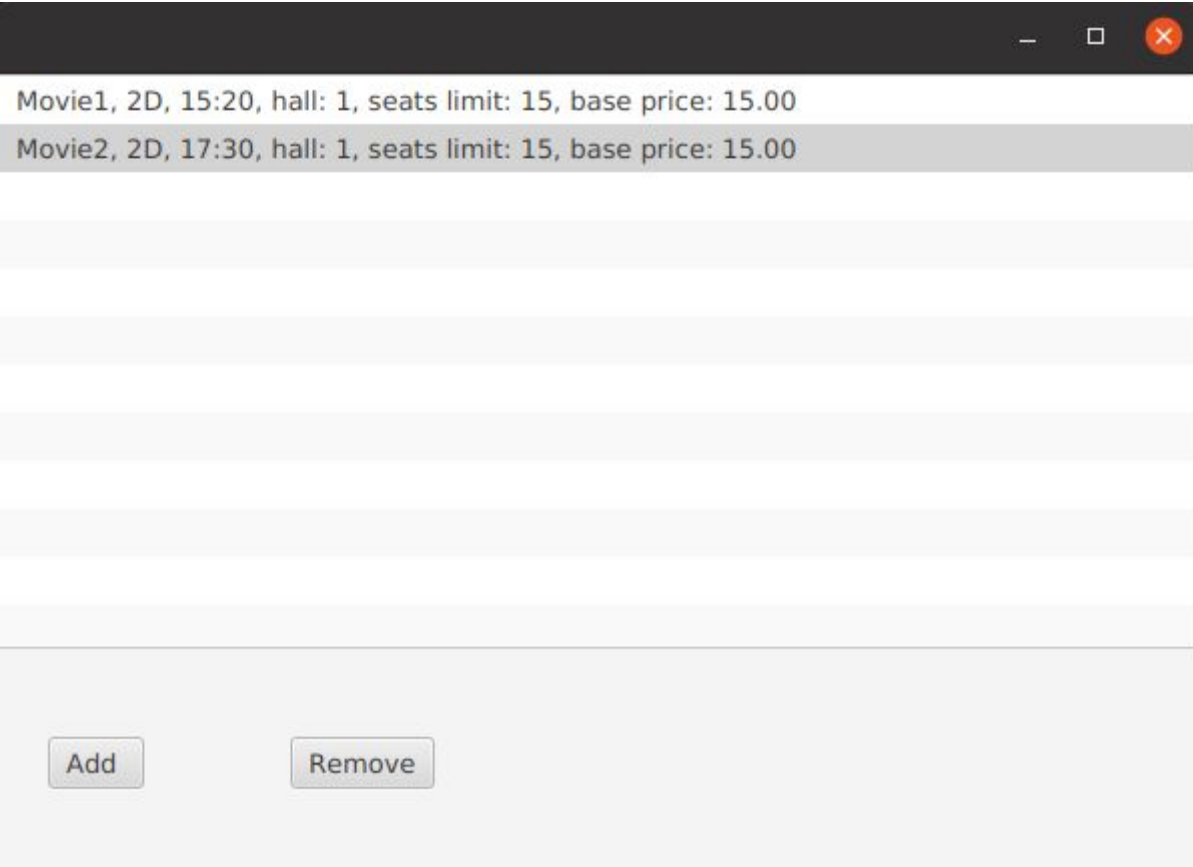
- *Lista filmów*

[illegible]

- *Panel pracownika*

A screenshot of a web application interface. At the top, there is a dark gray header bar containing three window control icons: a minus sign, a square, and a red circle with a white 'X'. The main content area is white and features three light gray buttons with rounded corners, arranged vertically in the center. The buttons are labeled 'Screenings', 'Tickets', and 'Logout' from top to bottom. The text on the buttons is in a dark gray, sans-serif font.

- *Lista seansów*



The screenshot shows a window titled "Lista seansów" with a dark header bar containing standard window controls (minimize, maximize, close). The main area is a list of movie showtimes, each on a separate row with alternating light and dark gray backgrounds. The first two rows contain data: "Movie1, 2D, 15:20, hall: 1, seats limit: 15, base price: 15.00" and "Movie2, 2D, 17:30, hall: 1, seats limit: 15, base price: 15.00". The remaining rows are empty. At the bottom of the window is a light gray bar containing two buttons: "Add" and "Remove".

Movie1, 2D, 15:20, hall: 1, seats limit: 15, base price: 15.00
Movie2, 2D, 17:30, hall: 1, seats limit: 15, base price: 15.00

Add Remove

- *Dodawanie seansów*

Movie

Date

Movie type

Time

Hall

Basic price

Add

- Dodawanie użytkowników

Login: adaber, Name: Adam, Surname: Bera

Login: addaac, Name: Ada, Surname: Bera

Login: grzjan, Name: Andrzej, Surname: Janosz

Login

Name

Surname

Add

Remove

Home

Logika

Zostały zrealizowane kontrolery dla widoków logowania, panelu administratora oraz zarządzania przez niego salami oraz filmami. Dodany został również kontroler do zarządzania użytkownikami, który nie został zrobiony wcześniej.

M3

1. Planowany progres

- Dopracowanie wyglądu
- Sortowanie po zadanych kryteriach
- Wyświetlanie repertuaru kina
- Statystyki
- Notyfikacje email

2. Podział obowiązków

- Grzegorz Janosz - Dopracowanie wyglądu (poprawienie prawie wszystkich widoków), Sortowanie, Statystyki, Email, Polecanie filmu, Poprawki w API persystencji
- Krzysztof Hardek - Widoki (przeznaczone do wizualnego dopracowania) oraz kontrolery dla: Zarządzania zniżkami (Admin), Funkcjonalności pracownika (Worker) oraz Funkcjonalności klienta (Customer)

3. Co zostało zrobione

Model

Model został zaktualizowany na potrzeby kolejnego kamienia milowego. Główne modyfikacje dotyczyły metod toString oraz do klasy Movies zostało dodane pole length przechowujące informację na temat długości filmu wyrażonej w minutach

Sortowanie, email, polecanie filmu, statystyki

Zostały stworzony odpowiedni interfejs do wysyłania maila na podany adres o dostarczonym temacie oraz treści. Zrealizowane zostały klasy implementujące interfejs Comparator, które umożliwiają nam sortowanie. Klasy dostarczające statystyki korzystają z wzorca Binding properties. Wystawiony został interfejs do polecania filmu użytkownikowi.

Widoki

Prawie wszystkie widoki zostały dopracowane, tak aby cały interfejs był bardziej przejrzysty oraz konsystentny.

Widoki dla niezalogowanych użytkowników

- Logowanie

Welcome to Cinema Multiplex!

Login

Type in your credentials

Register

Don't have an account?

- Rejestracja

Login

Name

Surname

Password

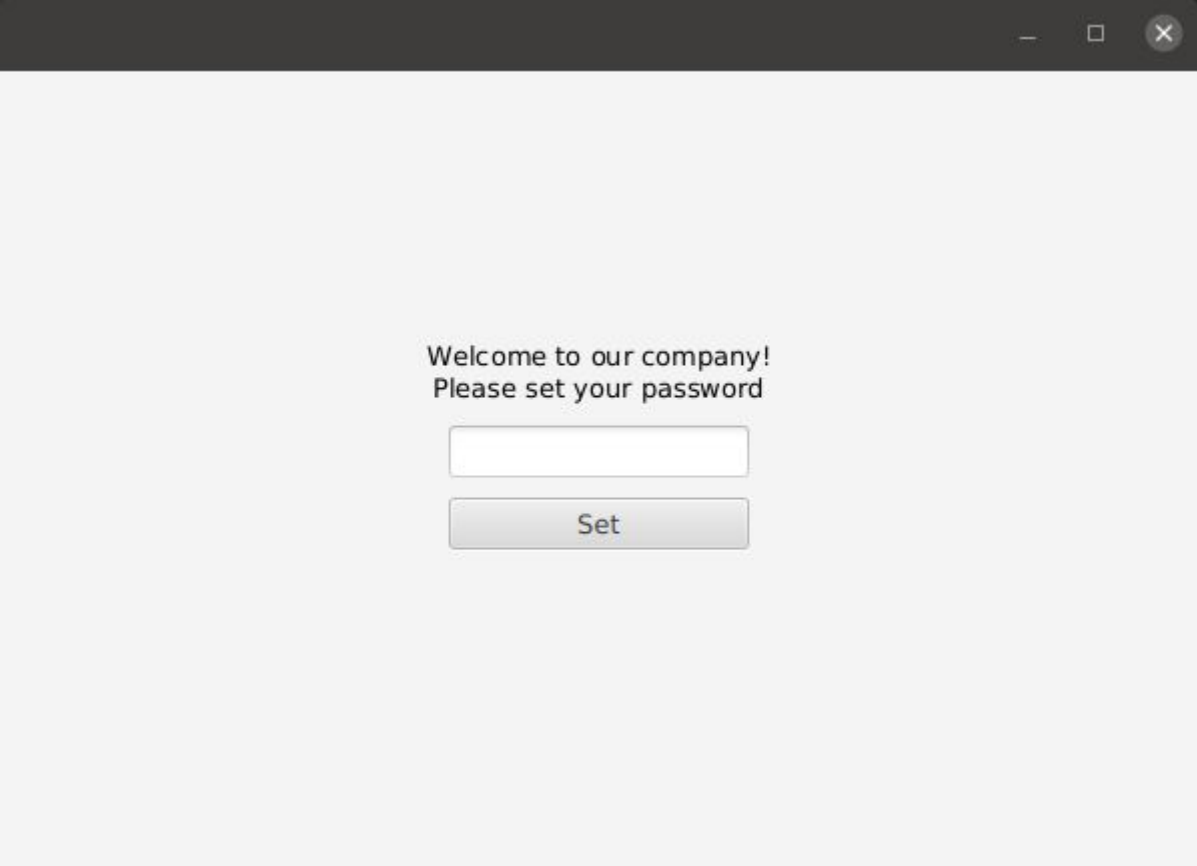
Date of birth

Email

Login

Register

- Ustawianie hasła dla nowego pracownika



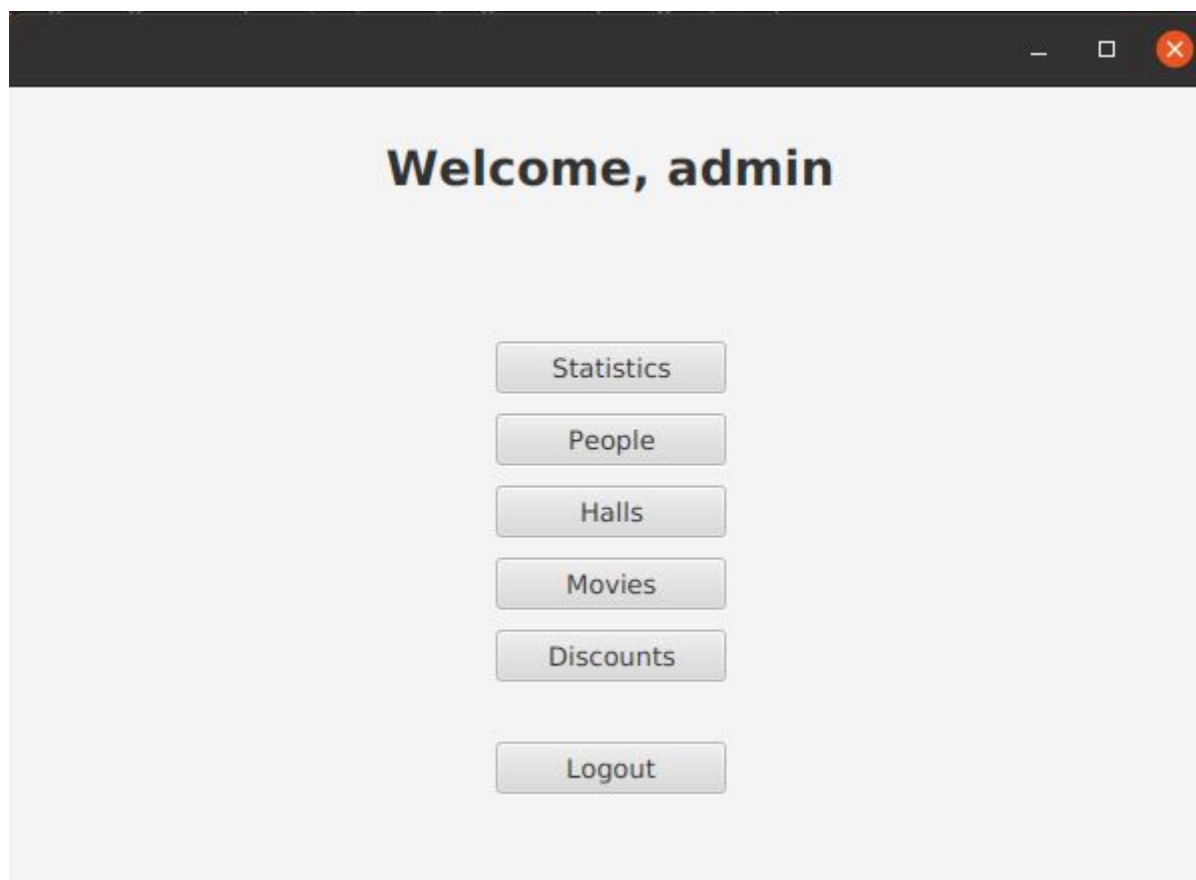
A screenshot of a web application window with a dark gray title bar containing standard window controls (minimize, maximize, close). The main content area is light gray and contains the following text and form elements:

Welcome to our company!
Please set your password

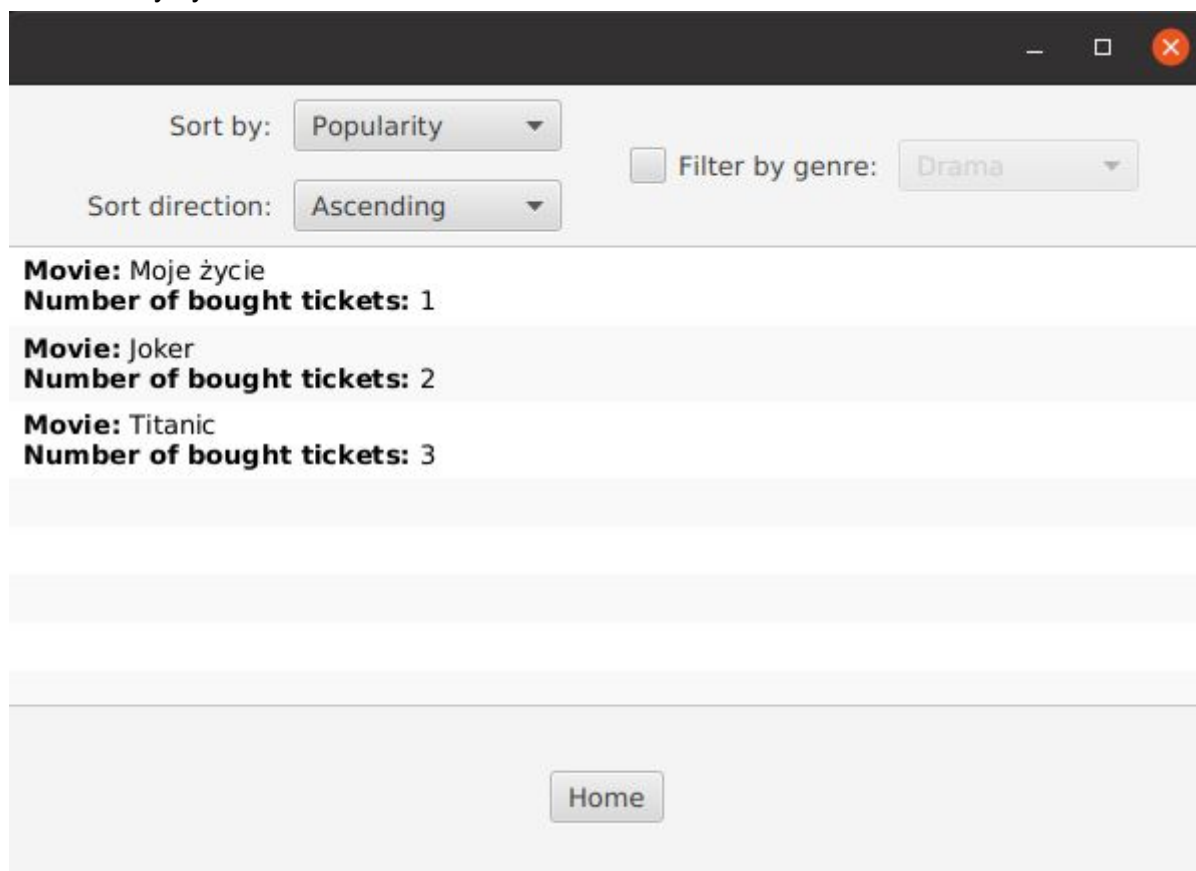
Below the text is a single-line text input field. Directly beneath the input field is a rectangular button with a light gray gradient and the word "Set" centered on it.

Widoki dla admina

- Panel główny



- Statystyki



- Lista użytkowników systemu

The screenshot shows a web application window with a dark title bar. The main content area has a light gray background. At the top right, there is a 'Sort:' label followed by a dropdown menu currently set to 'Login'. Below this, a list of user records is displayed, each on a new line. The records are: 'Login: admin, Name: Krzysztof, Surname: Hardek, Permissions: ADMIN', 'Login: emp, Name: emp, Surname: emp, Permissions: WORKER', 'Login: gregory, Name: Grzegorz, Surname: Janosz, gregory.janosz@gmail.com', 'Login: test, Name: test, Surname: test, Permissions: WORKER', 'Login: user, Name: user, Surname: user, user@user.pl', 'Login: user2, Name: user2, Surname: user2, user2@user.com', and 'Login: worker, Name: Andrzej, Surname: Basista, Permissions: WORKER'. Below the list, there is a form with four input fields: 'Login', 'Name', 'Surname', and 'Permissions' (which is a dropdown menu currently showing 'ADMIN'). To the right of these fields are two buttons: 'Add' and 'Remove'. At the bottom right of the form area is a 'Home' button.

Sort: Login

Login: admin, Name: Krzysztof, Surname: Hardek, Permissions: ADMIN
Login: emp, Name: emp, Surname: emp, Permissions: WORKER
Login: gregory, Name: Grzegorz, Surname: Janosz, gregory.janosz@gmail.com
Login: test, Name: test, Surname: test, Permissions: WORKER
Login: user, Name: user, Surname: user, user@user.pl
Login: user2, Name: user2, Surname: user2, user2@user.com
Login: worker, Name: Andrzej, Surname: Basista, Permissions: WORKER

Login
Name
Surname
Permissions

Add
Remove

Home

- Zarządzanie salami

The screenshot shows a web application window with a dark title bar. The main content area has a light gray background. It features a table with a blue border. The table has two columns: 'id' and 'seats limit'. The rows contain the following data: 'id: 0, seats limit: 40', 'id: 1, seats limit: 1', 'id: 10, seats limit: 50', 'id: 2, seats limit: 100', 'id: 3, seats limit: 50', 'id: 5, seats limit: 3', and 'id: 6, seats limit: 3'. Below the table, there is a form with two input fields: 'Hall id' and 'Seats limit'. To the right of these fields are two buttons: 'Add' and 'Remove'. At the bottom right of the form area is a 'Home' button.

id: 0, seats limit: 40
id: 1, seats limit: 1
id: 10, seats limit: 50
id: 2, seats limit: 100
id: 3, seats limit: 50
id: 5, seats limit: 3
id: 6, seats limit: 3

Hall id
Seats limit

Add
Remove

Home

- Zarządzanie filmami

Sort:

Title

Joker, 2020-12-27, genre: Thriller, length: 90

Moje życie , 2020-12-27, genre: Drama, length: 90

Shrek, 2021-01-12, genre: Drama, length: 90

Tenet, 2020-12-27, genre: Comedydrama, length: 150

Titanic, 2021-01-12, genre: Comedy, length: 194

Top gun, 2021-01-14, genre: Comedydrama, length: 90

Title

Date

Genre

Drama

Length

Add

Remove

Home

- Zarządzanie zniżkami

Seniorska , 15.00 %

Name

Value

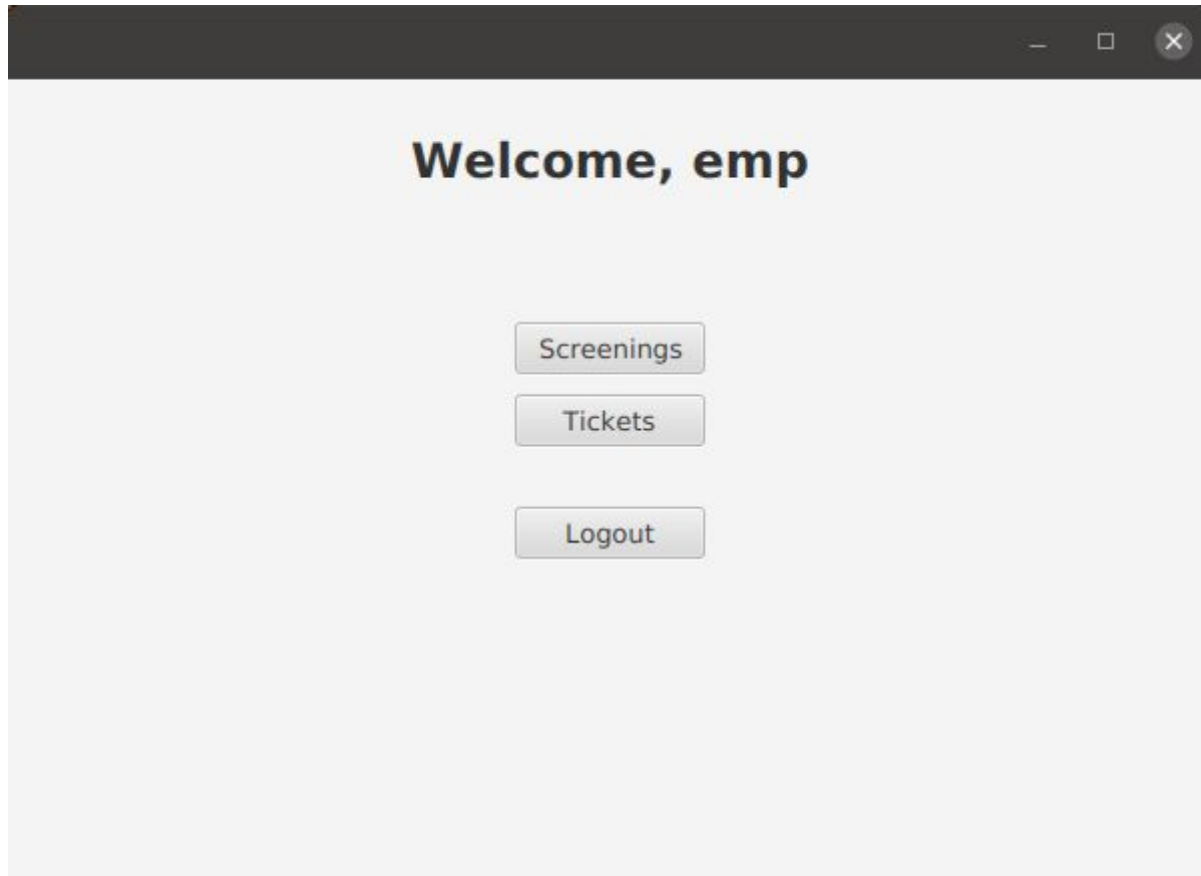
Add

Remove

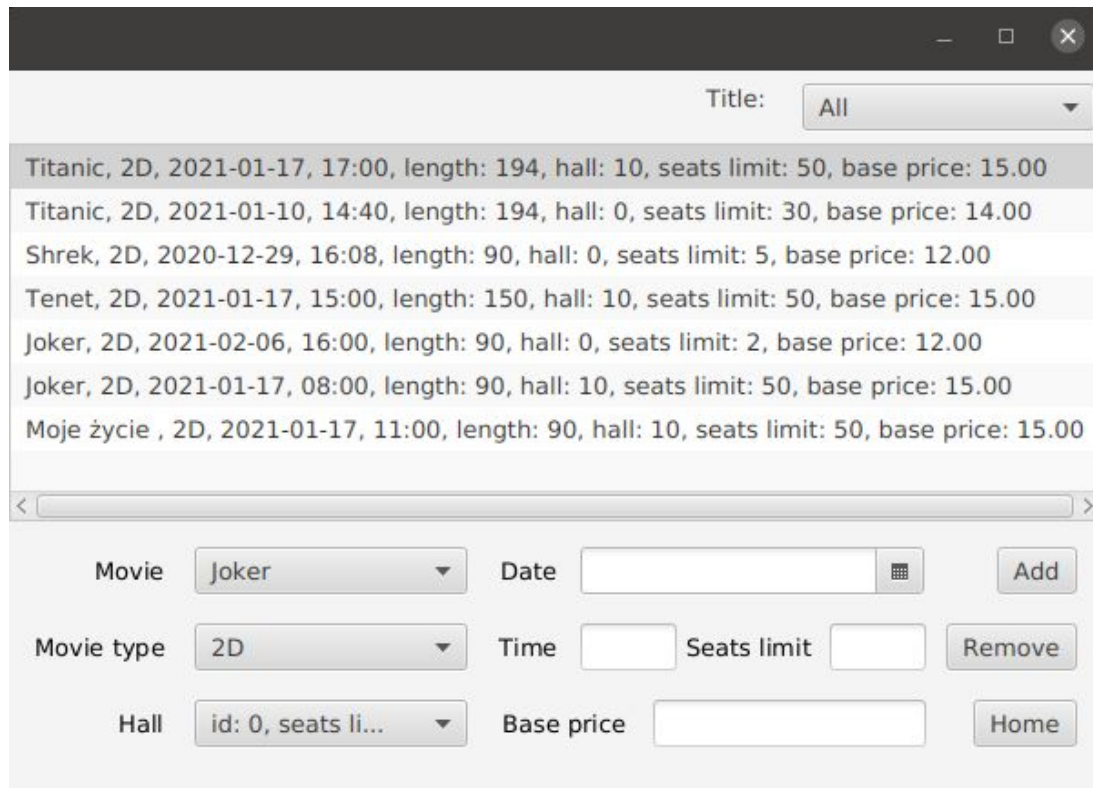
Home

Widoki dla zwykłego pracownika

- Panel główny



- Zarządzanie seansami

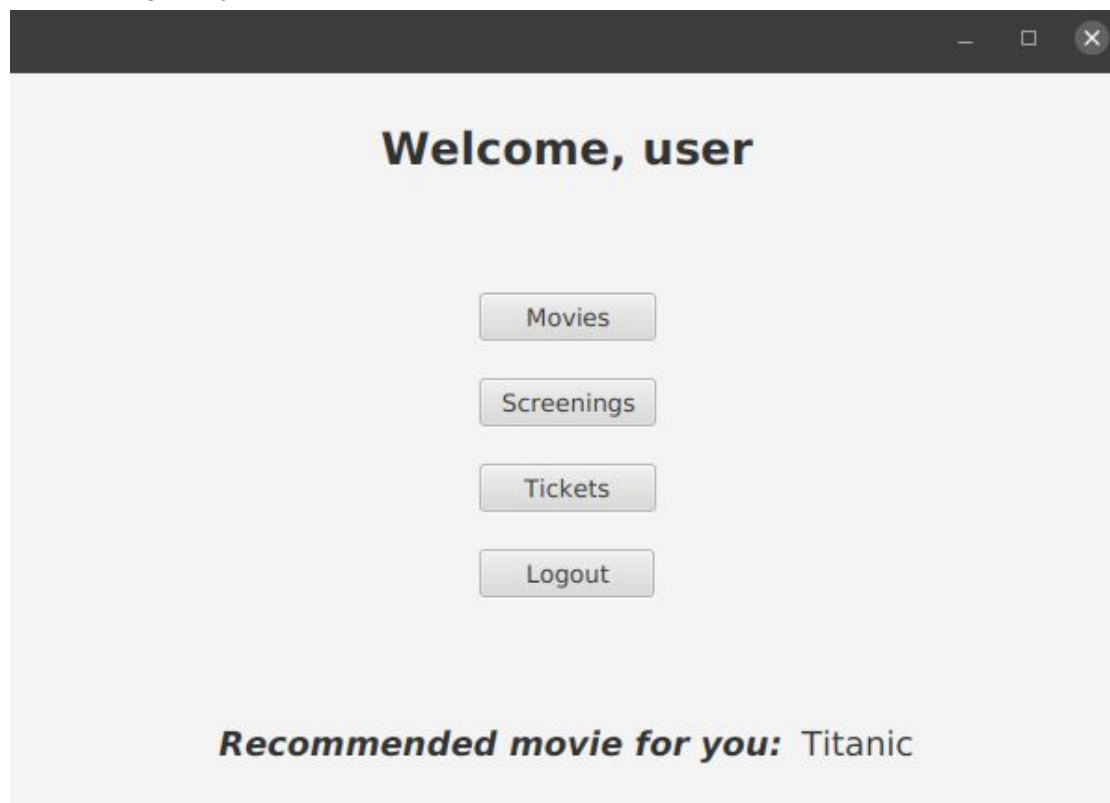


- Lista wszystkich biletów



Widoki dla klienta

- Panel główny



- Lista filmów

Sort:

Title

Joker, 2020-12-27, genre: Thriller, length: 90
Moje życie , 2020-12-27, genre: Drama, length: 90
Shrek, 2021-01-12, genre: Drama, length: 90
Tenet, 2020-12-27, genre: Comedydrama, length: 150
Titanic, 2021-01-12, genre: Comedy, length: 194
Top gun, 2021-01-14, genre: Comedydrama, length: 90

Screenings

Home

- Lista seansów

Title:

All

Titanic, 2D, 2021-01-17, 17:00, length: 194, hall: 10, seats limit: 50, base price: 15.00
Titanic, 2D, 2021-01-10, 14:40, length: 194, hall: 0, seats limit: 30, base price: 14.00
Shrek, 2D, 2020-12-29, 16:08, length: 90, hall: 0, seats limit: 5, base price: 12.00
Tenet, 2D, 2021-01-17, 15:00, length: 150, hall: 10, seats limit: 50, base price: 15.00
Joker, 2D, 2021-02-06, 16:00, length: 90, hall: 0, seats limit: 2, base price: 12.00
Joker, 2D, 2021-01-17, 08:00, length: 90, hall: 10, seats limit: 50, base price: 15.00
Moje życie , 2D, 2021-01-17, 11:00, length: 90, hall: 10, seats limit: 50, base price: 15.00

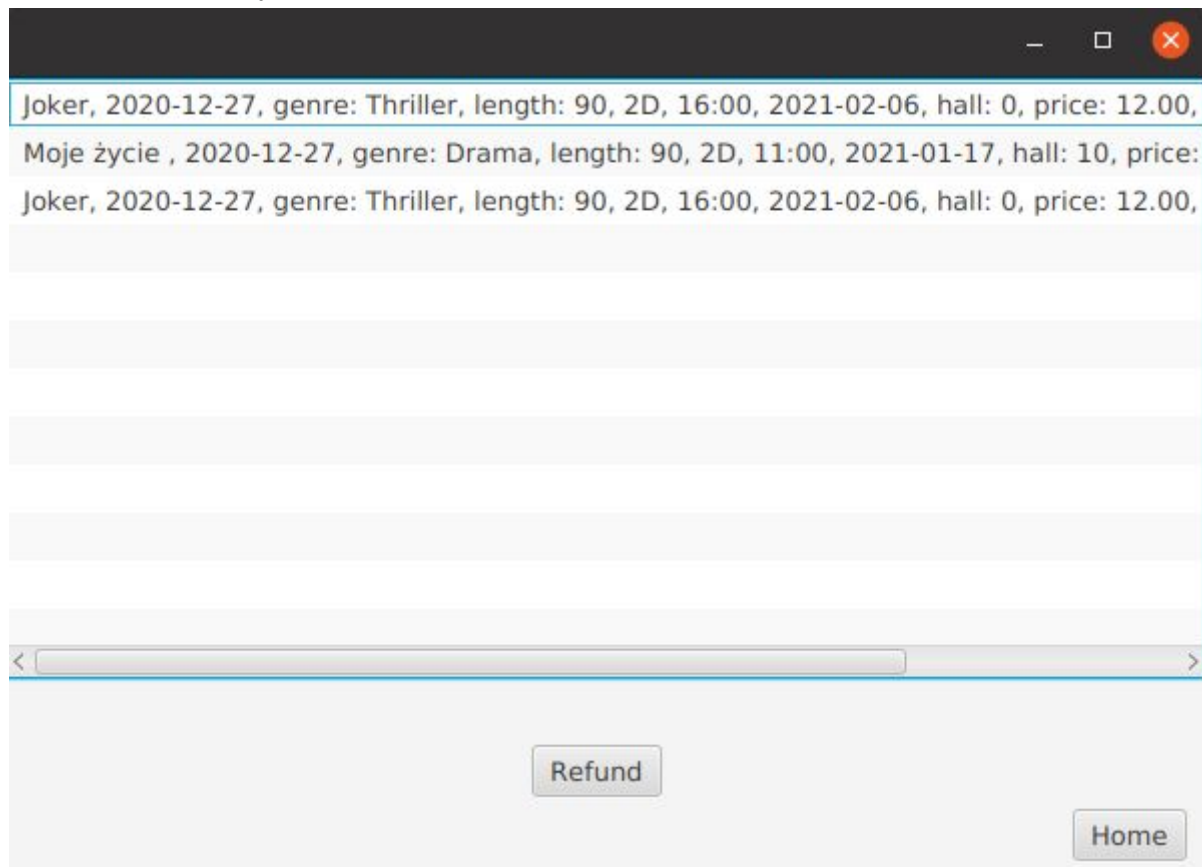
Discount:

None, 0.00 %

Buy ticket

Home

- Lista kupionych biletów



Logika

Zaimplementowane zostały kontrolery dla admina (zniżki oraz statystyki) oraz dla zwykłego użytkownika(wszystkie) i klienta(wszystkie). Stworzona została klasa Session, która reprezentuje sesję zalogowanego użytkownika. Realizuje ona wzorzec Singleton. Nasza aplikacja używa systemu walidacji formularzy. Został on stworzony z użyciem wzorca Composite. Każdy walidator zapisuje indywidualną wiadomość zwrotną na temat tego co poszło nie tak przy wpisywaniu danych. Informacje te są następnie prezentowane w graficznym interfejsie. Przy sortowaniu elementów list (np. w liście filmów) korzystamy z wzorca Observer. Umożliwia nam to dynamiczne aktualizowanie listy, przy zmianie klucza sortowania. Ponadto do tworzenia statystyk zostały użyte odpowiednie klasy generyczne oraz wzorzec predykat.