

Wypożyczalnia filmów

Karol Breguła

Spis treści

Specyfika problemu.....	2
Technologia.....	2
Struktura aplikacji.....	2
Hierarchia dziedziczenia klas.....	2
Charakterystyka pakietów i klas.....	3
Pakiet domyślny.....	4
Pakiet „model”.....	4
Client.....	4
ClientSet.....	5
CreditEntry.....	5
Order.....	5
OrderEntry.....	6
OrderSet.....	7
Person.....	7
Set<T>.....	7
Video.....	8
VideoSet.....	9
Pakiet „actions”.....	9
CloseAction.....	9
LoadAction.....	9
RemoveAction.....	9
SaveAction.....	9
SaveAllAction.....	9
AddOrderEntryAction.....	9
Pakiet „core”.....	9
Interface WindowObject<T>.....	9
ClientDialog.....	9
CreditEntryPicker.....	10
ExceptionDialog.....	10
ListModel<T>.....	10
OrderDialog.....	10
OrderEntryDialog.....	10
ResultSetDialog<T>.....	10
VideoDialog.....	10
VideoEntryPanel.....	10
Pakiet „tabs”.....	11
AbstractSetPanel.....	11
ClientPanel.....	11
OrderPanel.....	11
SearchOrderPanel.....	11
UtilsPanel.....	11
VideoPanel.....	11
Instrukcja użytkownika.....	11
Okno główne.....	11
Okno klienta.....	12

Okno wideo.....	12
Okno zamówienia.....	13
Widok wyszukiwania.....	14
Widok serwisowy.....	16

Specyfika problemu

Program obsługujący wypożyczalnię filmów. Obiektowy opis osób (obsada filmów oraz klienci wypożyczalni) i filmów oraz możliwość wiązania ze sobą tych informacji (wprowadzanie obsady filmów, reżysera, scenarzysty itd.). Zapamiętywanie historii wypożyczania dla filmów oraz klientów. Możliwość przeszukiwania i filtrowania bazy według zadanych kryteriów.

Przyjęto, że aplikacja będzie jednostaniskowa.

Technologia

Aplikacja została napisana z wykorzystaniem JavaSE-1.8 z wykorzystaniem biblioteki SWING. Dane są przechowywane w formie zserializowanych obiektów z wykorzystaniem interfejsu Serializable.

Struktura aplikacji

Hierarchia dziedziczenia klas

Występuje następująca hierarchia dziedziczenia klas:

- java.lang.Object
 - javax.swing.AbstractListModel<E> (implements javax.swing.ListModel<E>, java.io.Serializable)
 - core.ListModel<T> (implements java.lang.Iterable<T>)
 - model.Set<T> (implements java.lang.Iterable<T>)
 - model.ClientSet
 - model.OrderSet
 - model.VideoSet
 - actions.AddOrderEntryAction (implements java.awt.event.ActionListener)
 - Application
 - actions.CloseAction (implements java.awt.event.ActionListener)
 - java.awt.Component (implements java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable)
 - java.awt.Container
 - javax.swing.JComponent (implements java.io.Serializable)
 - javax.swing.JPanel (implements javax.accessibility.Accessible)
 - tabs.AbstractSetPanel
 - tabs.ClientPanel
 - tabs.OrderPanel
 - tabs.VideoPanel

- core.CreditEntryPicker
- tabs.SearchOrderPanel
- tabs.UtillsPanel
- core.VideoEntryPanel
- java.awt.Window (implements javax.accessibility.Accessible)
 - java.awt.Dialog
 - javax.swing.JDialog (implements javax.accessibility.Accessible, javax.swing.RootPaneContainer, javax.swing.WindowConstants)
 - core.ClientDialog (implements core.WindowObject<T>)
 - core.ExceptionDialog
 - core.OrderDialog (implements core.WindowObject<T>)
 - core.OrderEntryDialog (implements core.WindowObject<T>)
 - core.ResultsetDialog<T>
 - core.VideoDialog (implements core.WindowObject<T>)
- model.CreditEntry (implements java.io.Serializable)
- core.CreditEntryAddAction (implements java.awt.event.ActionListener)
- actions.LoadAction (implements java.awt.event.ActionListener)
- model.Order (implements java.io.Serializable)
- model.OrderEntry (implements java.io.Serializable)
- tester.OrderTester (implements tester.SearchTester<T>)
- model.Person (implements java.io.Serializable)
 - model.Client (implements java.io.Serializable)
- actions.RemoveAction (implements java.awt.event.ActionListener)
- actions.SaveAction (implements java.awt.event.ActionListener)
- actions.SaveAllAction (implements java.awt.event.ActionListener)
- actions.SearchAction<T> (implements java.awt.event.ActionListener)
- actions.UpdateObjectAction (implements java.awt.event.ActionListener)
- actions.UpdateObjectListAction<T> (implements java.awt.event.ActionListener)
- model.Video (implements java.io.Serializable)

Charakterystyka pakietów i klas

Aplikacja została podzielona na następujące pakiety:

- actions,
- core,

- model,
- tabs,
- pakiet domyślny.

Pakiet domyślny

Pakiet domyślny zawiera wyłącznie klasę „Application”, która zawiera odesłanie do głównych elementów aplikacji i inicjuje bazę danych. Klasa rozszerza `java.lang.Object`.

Klasa posiada następujące pola prywatne:

- `ClientSet` `clientset`
- `OrderSet` `orderset`
- `VideoSet` `videoset`

Pola te są związane z inicjowaniem obiektów bazy danych. Zdecydowano się nie wykorzystywać wzorca Singleton do bazy danych, albowiem Singleton jest określany antywzorcem ze względu na rodzący problemy analogiczne do zmiennych globalnych w programowaniu funkcyjnym.

Klasa posiada konstruktor, które nie przyjmuje żadnych parametrów i klasę `main`, która nie przyjmuje żadnych dodatkowych parametrów.

Dane prezentowane są z wykorzystaniem zakładek zgromadzonych w pakiecie „tabs”.

Pakiet „model”

Niniejszy pakiet gromadzi klasy, których przeznaczeniem jest zarządzanie danymi, ale nie operacje na danych.

Są to:

- `Client`
- `ClientSet`
- `CreditEntry`
- `Order`
- `OrderEntry`
- `OrderSet`
- `Person`
- `Set<T>`
- `Video`
- `VideoSet`

Client

Przeznaczeniem klasy `Client` jest reprezentowanie indywidualnego klienta korzystającego z

wypożyczalni.

Klasa Client rozszerza model.Person.

ClientSet

Przeznaczeniem klasy ClientSet jest reprezentowanie zbioru klientów.

Klasa implementuje java.io.Serializable, java.lang.Iterable<Client>, javax.swing.ListModel<Client> i model.Set<Client>.

Klasa dziedziczy z model.Set<Client>, które dziedziczy z core.ListModel<T>, które dziedziczy z javax.swing.AbstractListModel<T>.

Klasa nie implementuje samodzielnie metod.

CreditEntry

Przeznaczeniem klasy jest reprezentowanie wpisu opisującego dane obsady filmowej przy karcie filmu.

Publiczna klasa CreditEntry rozszerza java.lang.Object i implementuje java.io.Serializable

Klasa implementuje następujące metody:

- toString() – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika.

Posiada dwa konstruktory:

- public CreditEntry(java.lang.String firstName, java.lang.String secondName, java.lang.String role)
- public CreditEntry(Person person, java.lang.String role)

Order

Przeznaczeniem klasy „Order” jest gromadzenie danych na temat zamówienia.

Publiczna klasa Order rozszerza java.lang.Object i implementuje java.io.Serializable.

Klasa posiada dwa publiczne konstruktory:

- Order()
- Order(Client client, ListModel<OrderEntry> data, java.util.Date createdOn)

Klasa posiada następujące pola:

- private Client client – referencja do obiektu, który dokonał zamówienia
- private ListModel<OrderEntry> data – kolekcja pozycji zamówienia,
- private Date createdOn – data utworzenia zamówienia
- private Date returndate – data zwrotu

Klasa implementuje następujące metody:

- int getAge() – metoda zwraca wiek zamówienia tj. różnicę między czasem utworzenia i

zwrotu, ewentualnie w przypadku nieustalonego czasu zwrotu różnicę między czasem utworzenia, a czasem aktualnym.

- `boolean hasVideo(Video video)` – metoda określa czy zamówienie obejmuje wskazane wideo,
- `java.lang.String toString()` – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika.
- `java.math.BigDecimal TotalValue()` – metoda, która oblicza na podstawie posiadanej kolekcji sumaryczną wartość zamówienia.

Klasa implementuje także następujące mutatory:

- `Client getClient()`
- `java.util.Date getCreatedOn()`
- `ListModel<OrderEntry> getData()`,
- `java.util.Date getReturnDate()`
- `void setClient(Client client)`
- `void setCreatedOn(java.util.Date createdOn)`
- `void setData(ListModel<OrderEntry> data)`
- `void setReturndate(java.util.Date returndate)`

OrderEntry

Przeznaczeniem klasy „OrderEntry” jest reprezentowanie pozycji konkretnego zamówienia.

Publiczna klasa OrderEntry rozszerza `java.lang.Object` i implementuje `java.io.Serializable`.

Klasa posiada pola prywatne „video” typu `Video`.

Zważywszy na charakter pracy wypożyczalni zdecydowano się na odstępianie od przechowywania dodatkowych informacji np. określających ilość wypożyczonych filmów. Kto wypożycza w wypożyczalni kilka razy jednocześnie ten sam film? Bezsens.

Klasa posiada dwa konstruktory:

- `public OrderEntry(Video video)`
- `public OrderEntry()`

Klasa implementuje następujące metody:

- `java.lang.String toString()` – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika.

Klasa posiada następujące mutatory:

- `Video getVideo()`

- void setVideo(Video video)

OrderSet

Przeznaczeniem klasy „OrderSet” jest reprezentowanie zbioru zamówień („Order”).

Publiczna public class OrderSet dziedziczy z Set<Order> i implementuje następujące interfejsy java.io.Serializable, java.lang.Iterable<Order>, javax.swing.ListModel<Order>.

Klasa implementuje następujące metody:

- java.lang.String toString() – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika.

Person

Przeznaczeniem klasy jest reprezentowanie osób (obsada filmów oraz klienci wypożyczalni).

Klasa posiada konstruktor:

- Person(java.lang.String firstName, java.lang.String secondName)

Klasa implementuje następujące metody:

- java.lang.String toString() – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika,
- java.lang.String getName() – metoda, która umożliwia tekstową samego imienia i nazwiska bez innych danych opisujących obiekt.

Klasa posiada następujące mutatory:

- java.lang.String getFirstName()
- java.lang.String getSecondName()
- void setFirstName(java.lang.String firstName)
- void setSecondName(java.lang.String secondName)

Set<T>

Przeznaczeniem klasy Set<T> jest dostarczenie abstrakcyjnego zbiorów obiektów, których sprawnie będzie się odbywał zapis i inne operacje związane z plikami.

Klasa dziedziczy z extends ListModel<T> i implementuje java.io.Serializable, java.lang.Iterable<T>, javax.swing.ListModel<T>

Klasa implementuje następujące metody:

- boolean add(T e) – dodaje obiekt do zbioru
- void clear() – czyści zbiór
- void fromFile(java.io.ObjectInputStream input) – wczytuje dane z strumienia do zbioru,
- T get(int index) – zwraca obiekt o wskazanym indeksie,

- `java.util.Iterator<T> iterator()` – implementacja wzorca delegata, zapewniająca wsparcie dla poprawnego iterowania przez obiekt,
- `int size()` – zwraca ilość obiektów zgromadzonych w zbiorze,
- `void toFile(java.io.ObjectOutputStream output)` – zapisuje dane zgromadzone w zbiorze do strumienia,
- `java.lang.String toString()` – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika.

Video

Przeznaczeniem klasy jest Video jest reprezentowanie indywidualnego filmu dostępnego w wypożyczalni.

Publiczna klasa Video rozszerza `java.lang.Object` implementuje `java.io.Serializable`.

Klasa posiada następujące konstruktory:

- `public Video()`
- `public Video(java.lang.String title, int diskTotal, int diskFree, java.math.BigDecimal perDay)`
- `public Video(java.lang.String title, ListModel<CreditEntry> credits, int diskTotal, int diskFree, java.math.BigDecimal perDay)`

Klasa posiada następujące mutatory:

- `ListModel<CreditEntry> getCredits()`
- `int getDiskFree()`
- `int getDiskTotal()`
- `java.math.BigDecimal getPerDay()`
- `java.math.BigDecimal getPrice(int day)`
- `java.lang.String getTitle()`
- `void setDiskFree(int diskFree)`
- `void setDiskTotal(int diskTotal)`
- `void setPerDay(java.math.BigDecimal perDay)`
- `void setTitle(java.lang.String title)`

Klasa implementuje następujące metody:

- `java.lang.String toString()` – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika.

VideoSet

Przeznaczeniem klasy VideoSet jest reprezentowanie zbioru wszystkich filmów dostępnych w wypożyczalni.

Publiczna klasa VideoSet rozszerza Set<Video> i implementuje następujące interfejsy java.io.Serializable, java.lang.Iterable<Video>, javax.swing.ListModel<Video>

Klasa implementuje następujące metody:

- java.lang.String toString() – metoda, która umożliwia tekstową reprezentację obiektu dla użytkownika.

Pakiet „actions”

Pakiet gromadzi klasy implementujące java.awt.event.ActionListener.

CloseAction

Klasa odpowiedzialna za akcję zamknięcia aktywnego okna.

LoadAction

Klasa odpowiedzialna za akcję wczytania danych z wybranego pliku do określonego zbioru.

RemoveAction

Klasa odpowiedzialna za akcję usunięcia aktywnego elementu listy.

SaveAction

Klasa odpowiedzialna za akcję zapisania określonego zbioru do wybranego pliku.

SaveAllAction

Klasa odpowiedzialna za akcję zapisania określonych zbiorów do oddzielnych plików w wybranym katalogu.

AddOrderEntryAction

Klasa odpowiedzialna za akcję dodania do modelu pozycji zamówienia.

Pakiet „core”

Pakiet gromadzi klasy i interfejs operacji na danych i okien.

Interface WindowObject<T>

Interfejs, który wzbogaca okno o operacje wypisania obiektu operacji okna.

Intefejs wymusza metody:

- T getObject() – zwraca obiekt, który był przetwarzany w oknie lub został utworzony
- boolean getStatus() – zwraca status okna tj. czy użytkownik zaakceptował zmiany treści pól.

ClientDialog

Klasa odpowiedzialna za okno operacji na obiekcie typu „klient” tj. wczytanie danych do pól,

wyświetlenie okna i zapisanie danych ponownie do obiektu.

Publiczna klasa ClientDialog rozszerza javax.swing.JDialog implementując WindowObject<Client>.

CreditEntryPicker

Klasa odpowiedzialna za okno stworzenia obiektu „CreditEntryPicker” na podstawie danych wskazanych przez użytkownika.

Publiczna klasa CreditEntryPicker rozszerza javax.swing.JPanel

ExceptionDialog

Klasa odpowiedzialna za okno wskazujące treść zaistniałego wyjątku.

Publiczna klasa ExceptionDialog rozszerza javax.swing.JDialog.

ListModel<T>

Klasa odpowiedzialna za dostarczenie implementacji „modelu” listy dla obiektu Jlist na podstawie listy List np. ArrayList, LinkedList.

Publiczna klasa ListModel<T> rozszerza javax.swing.AbstractListModel<T> implementując java.lang.Iterable<T>

OrderDialog

Klasa odpowiedzialna za okno operacji na zamówieniu umożliwiając ustawienie wszystkich niezbędnych atrybutów.

Publiczna klasa OrderDialog rozszerza javax.swing.JDialog implmentując WindowObject<Order>

OrderEntryDialog

Klasa odpowiedzialna za okno operacji na pozycji zamówienia umożliwiając jego utworzenie, usuwanie itd.

Publiczna klasa OrderEntryDialog rozszerza javax.swing.JDialog implementując WindowObject<OrderEntry>

ResultSetDialog<T>

Klasa odpowiedzialna za okno wyników wyszukiwania.

Publiczna klasa ResultSetDialog<T> rozszerza javax.swing.JDialog

VideoDialog

Klasa odpowiedzialna za okno operacji na wideo umożliwiając jego utworzenie, ewentualnie edycje.

Publiczna klasa VideoDialog rozszerza javax.swing.JDialog implementując WindowObject<Video>.

VideoEntryPanel

Klasa odpowiedzialna za panel edycji obsady wideo umożliwiając utworzenie i usuwanie pozycji.

Publiczna klasa VideoEntryPanel rozszerza javax.swing.JPanel.

Pakiet „tester”

Interfejs SearchTester

Interfejs zapewniający implementacje weryfikacji wyniku wyszukiwania.

Klasa OrderTester

Klasa implementująca w/w interfejs wyszukiwania dla przeszukiwania zamówienia według zadanych kryteriów.

Pakiet „tabs”

Niniejsza pakiet gromadzi panele odpowiedzialne zakładek, które odpowiadają za podstawowe funkcjonalności niniejszego oprogramowania.

AbstractSetPanel

Panel odpowiedzialny za dostarczenie podstawowych, generycznych cech widoku zbioru tj. usuwanie, zapis, odczyt.

Publiczna abstrakcyjna klasa AbstractSetPanel rozszerza javax.swing.JPanel. Podklasy to ClientPanel, OrderPanel, VideoPanel.

ClientPanel

Panel odpowiedzialny za dostarczenie podstawowych operacji zmian zbioru klientów.

Publiczna klasa rozszerzająca AbstractSetPanel.

OrderPanel

Panel odpowiedzialny za dostarczenie podstawowych operacji zmian zbioru zamówień.

Publiczna klasa rozszerzająca AbstractSetPanel.

SearchOrderPanel

Panel odpowiedzialny za dostarczenie operacji przeszukiwania zbioru zamówień według zadanych kryteriów tj. filmu występującego lub klienta.

Wynik operacji prezentowany jest z wykorzystaniem ResultSetDialog<T>.

Publiczna klasa rozszerzająca AbstractSetPanel.

UtilsPanel

Panel odpowiedzialny za dostarczenie zróżnicowanych akcji serwisowych aplikacji. Dotychczas jest to zapisanie jednocześnie wszystkich zbiorów aplikacji.

VideoPanel

Panel odpowiedzialny za dostarczenie podstawowych operacji zmian zbioru wideo

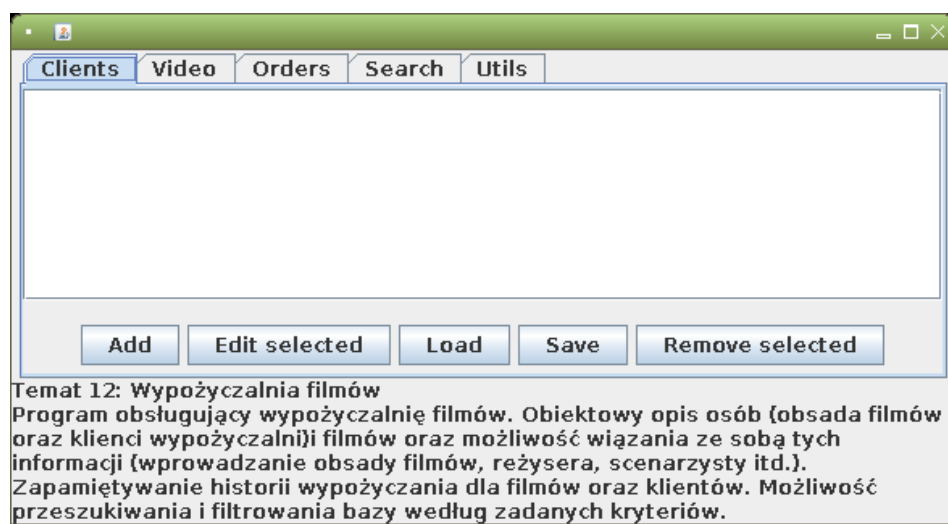
Publiczna klasa rozszerzająca AbstractSetPanel.

Instrukcja użytkowania

Aplikacja posiada szereg metod „main” pozwalając na wyświetlenie poszczególnych okien w sposób niezależny, autonomiczny, jednak ze względu na zakres operacji zaleca się uruchomienie z wykorzystaniem Application.java.

Okno główne

B ezpośrednio po uruchomieniu programu zgodnie z powyższym otrzymujemy:



Ilustracja 1: Okno główne

Poszczególne zakładki umożliwiają:

- Clients – operacje na zbiorze klientów,
- Video – operacje na zbiorze wideo,
- Orders – operacje na zbiorze zamówień,
- Search – przeszukiwanie bazy danych według zadanych kryteriów.
- Utils – wykonanie operacji serwisowych.

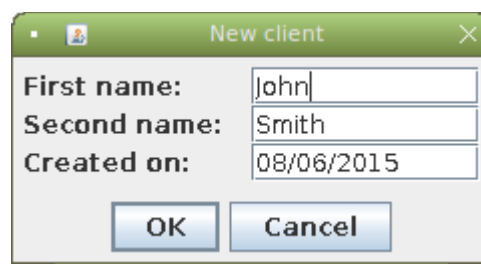
Okno klienta

Po wybraniu „Add” w zakładce „Clients” wywołujemy okno operacji nad klientem.

Wprowadzone są pewne początkowe dane, które jednak można swobodnie zmieniać.

Wybranie „OK” powoduje dopisanie danych do bazy aplikacji i wyświetlenie monitu dla użytkownika.

Wybranie „Cancel” anuluje operację.



Ilustracja 2: Okno klienta

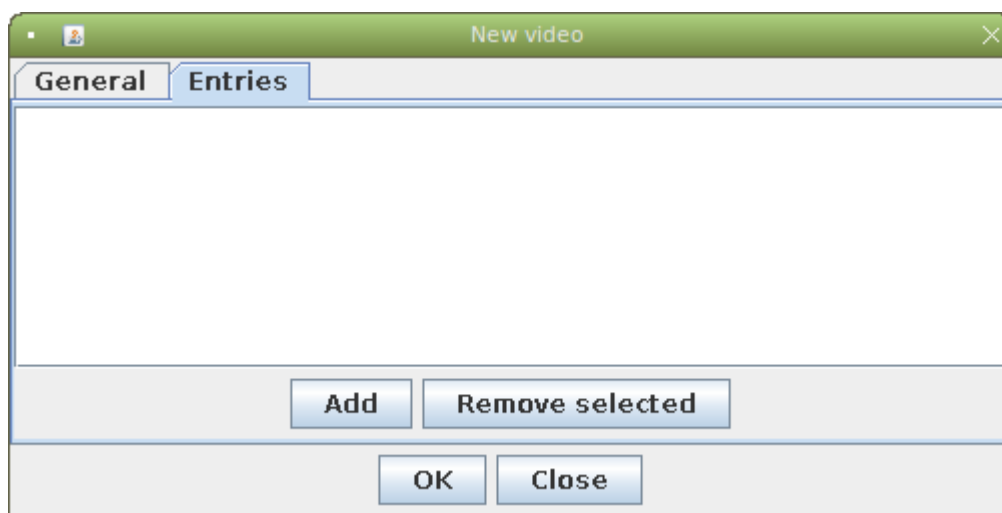
Okno wideo

W zakładce „Wideo” po wybraniu „Add” lub „Edit selected” (pod warunkiem wybrania obiektu edycji) możliwe jest wywołanie okna wideo.

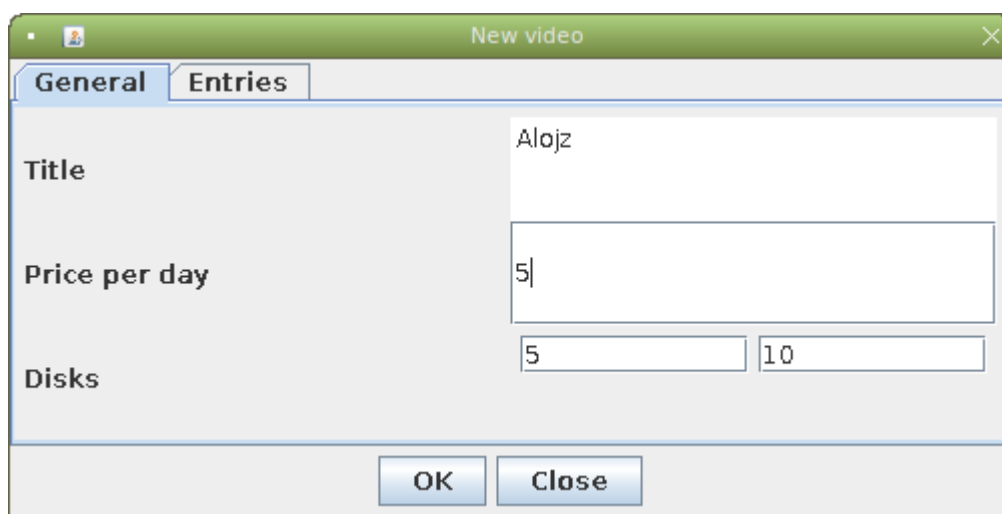
Okno wideo służy do edycji danych wideo przechowywanych w aplikacji.

Zbudowane jest z dwóch zakładek:

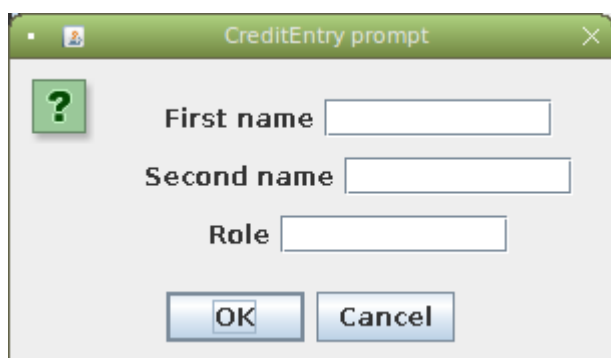
- General – przedstawia podstawowe informacje o wideo (tytuł, cenę za dzień wypożyczenia, stan nośników),
- Entries – przedstawia obsadę filmu umożliwiając usuwanie i dodawanie rekordów.



Ilustracja 4: Okno wideo - zakładka "Entries"



Ilustracja 5: Okno wideo - zakładka "General"



Ilustracja 6: Okno obsady

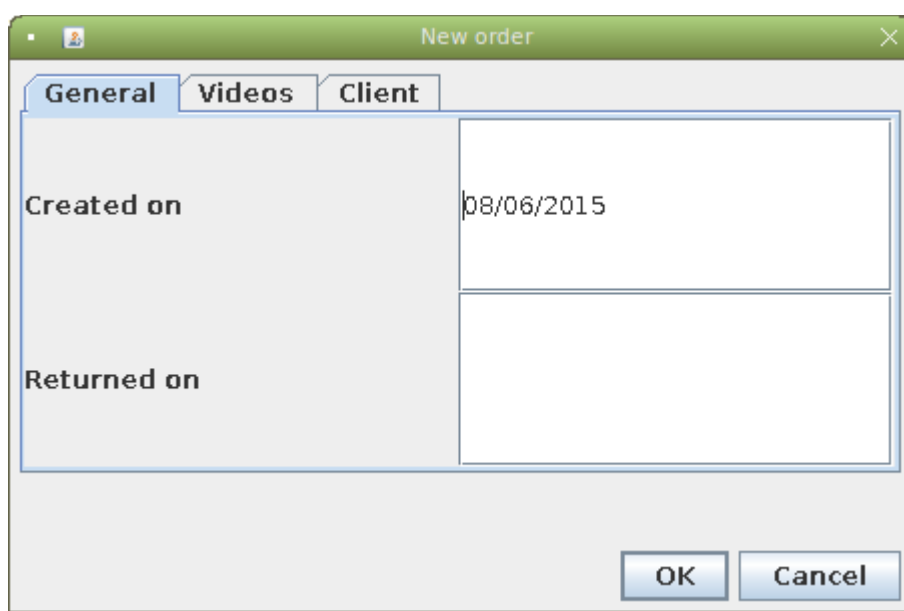
Okno zamówienia

W zakładce „Order” po wybraniu „Add” lub „Edit selected” (pod warunkiem wybrania obiektu edycji) możliwe jest wywołanie okna zamówienia.

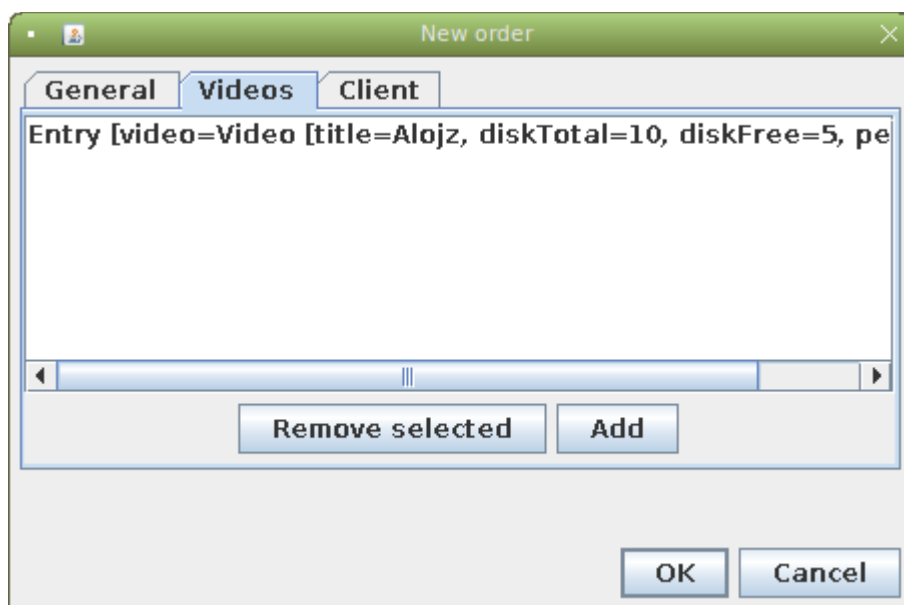
Okno zamówienia umożliwia utworzenie lub edycje zamówienia.

Okno składa się z trzech zakładek:

- General – umożliwia ustalenie podstawowych danych zamówienia tj. czas utworzenia i zmiany,
- Videos – umożliwia zmianę wideo objętych zamówieniem, w tym dodanie pozycji spośród bazy aplikacji,
- Client – umożliwia wybranie spośród bazy aplikacji, który dokonał zamówienia.

The screenshot shows a window titled "New order" with a green header bar. It has three tabs: "General", "Videos", and "Client". The "General" tab is selected. Inside the tab, there are two labels: "Created on" and "Returned on". The "Created on" field contains the date "08/06/2015". The "Returned on" field is empty. At the bottom right, there are "OK" and "Cancel" buttons.

Ilustracja 7: Okno zamówienia - zakładka "General"

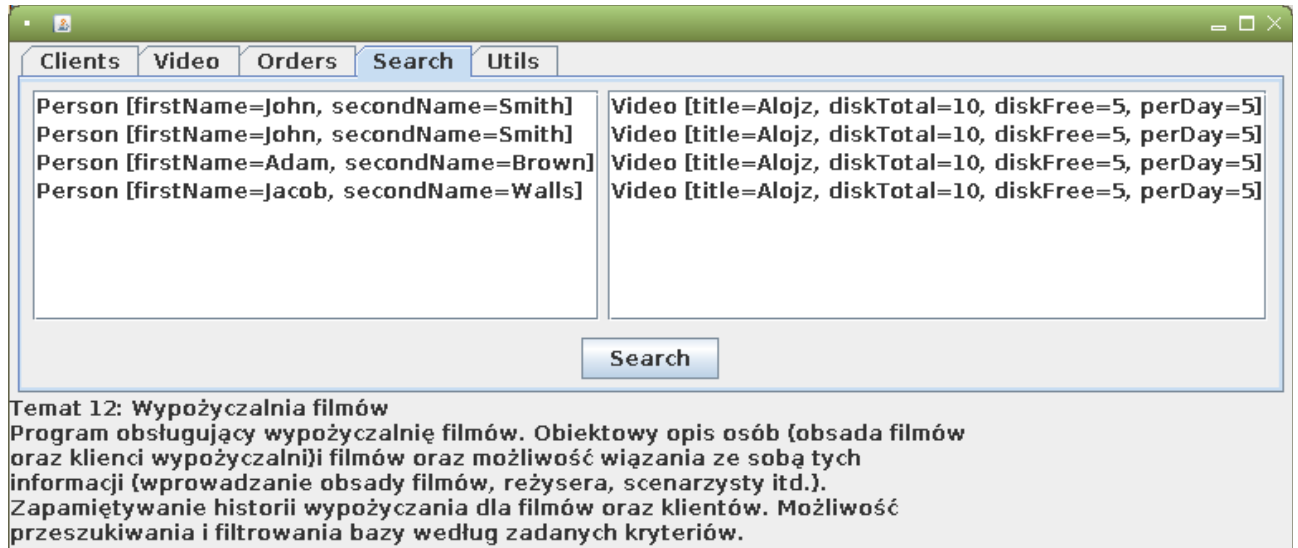
The screenshot shows the same "New order" window, but with the "Videos" tab selected. The "General" tab is now disabled. The main area of the "Videos" tab contains a text field with the text "Entry [video=Video [title=Alojz, diskTotal=10, diskFree=5, pe". Below this text field is a horizontal scrollbar. At the bottom of the tab area, there are "Remove selected" and "Add" buttons. At the bottom right of the window, there are "OK" and "Cancel" buttons.

Ilustracja 8: Okno zamówienia - zakładka "Videos"

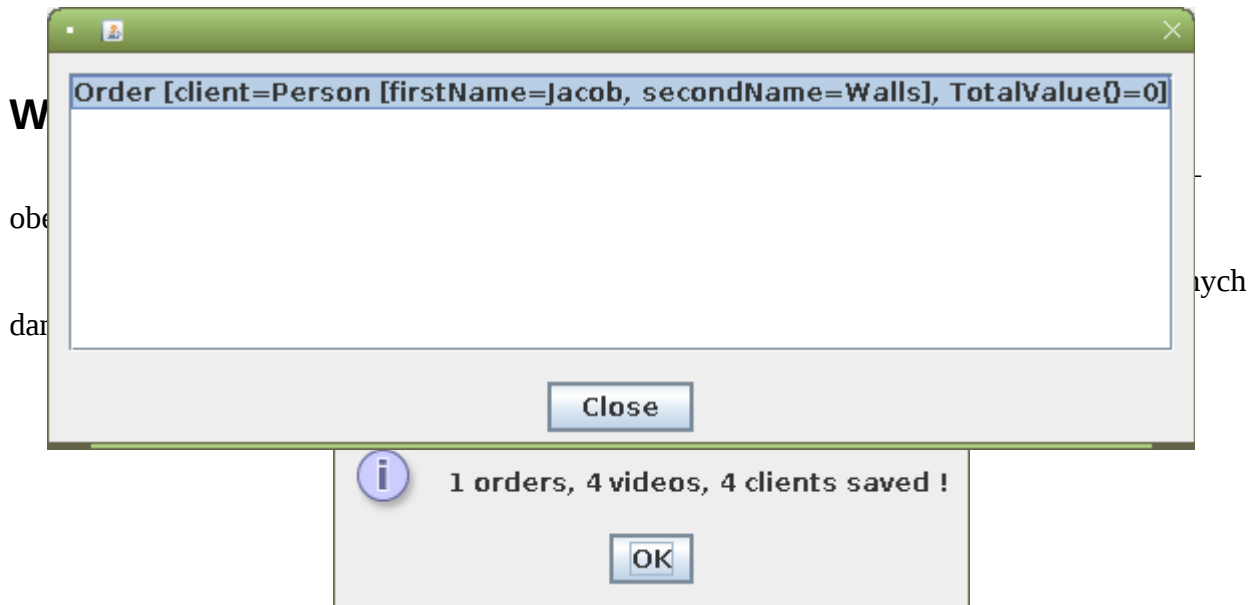
Widok wyszukiwania

W zakładce „Search” okna głównego dostępny jest ekran wyszukiwania. Umożliwia on przeszukiwanie bazy danych według zadanych kryteriów. Nie wybranie danego kryterium prowadzi do jego zignorowania.

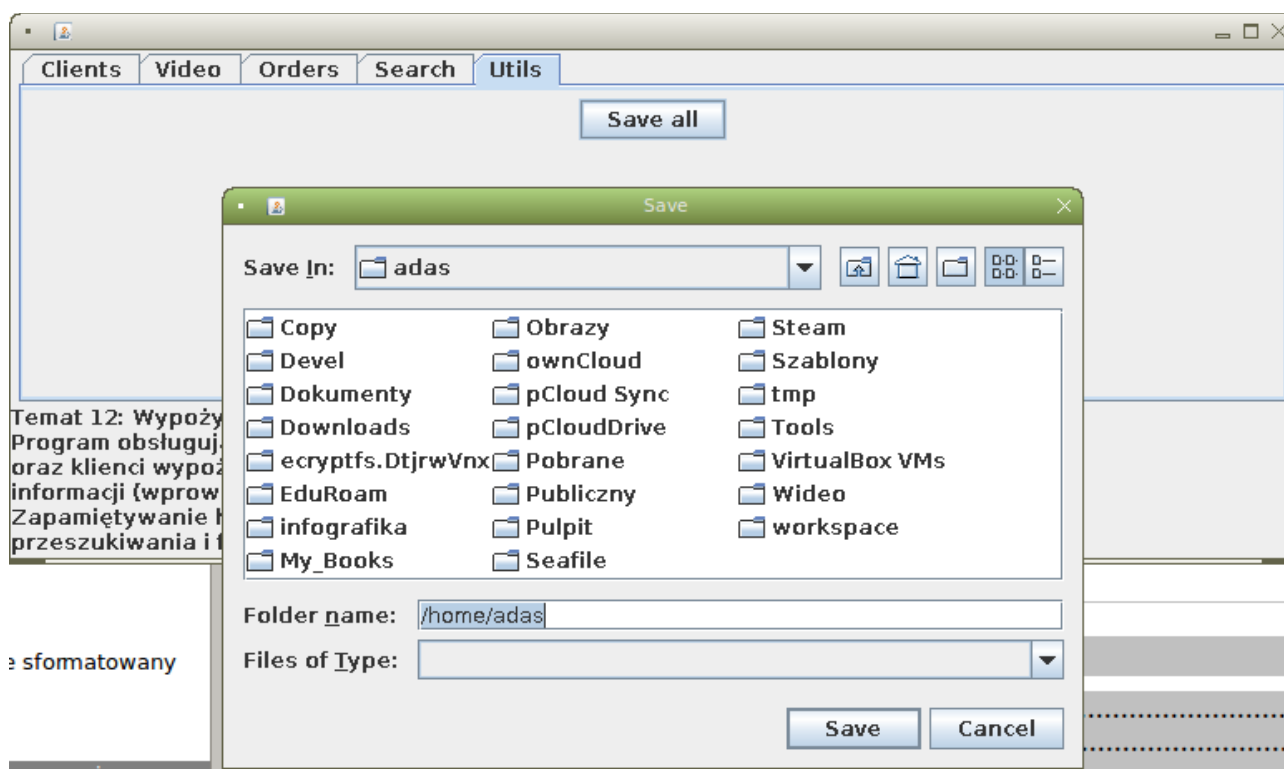
Z jednej strony dostępna jest baza klientów, a z drugiej strony dostępna jest baza klientów.



Ilustracja 9: Widok wyszukiwania



Ilustracja 11: Monit zapisu wielu zbiorów



Ilustracja 12: Okno wyboru lokalizacji zapisu zbiorów

Indeks ilustracji

Ilustracja 1: Okno główne.....	11
Ilustracja 2: Okno klienta.....	12
Ilustracja 3: Monit zapisania klienta.....	12
Ilustracja 4: Okno wideo - zakładka "Entries".....	13
Ilustracja 5: Okno wideo - zakładka "General".....	13
Ilustracja 6: Okno obsady.....	13
Ilustracja 7: Okno zamówienia - zakładka "General".....	14
Ilustracja 8: Okno zamówienia - zakładka "Videos".....	14
Ilustracja 9: Widok wyszukiwania.....	15
Ilustracja 10: Okno wyników wyszukiwania.....	15
Ilustracja 11: Monit zapisu wielu zbiorów.....	15
Ilustracja 12: Okno wyboru lokalizacji zapisu zbiorów.....	16