

<b>AI1</b>	<b>Dokumentacja projektu</b>
<b>Autor</b>	Karol Bał, 123630
<b>Kierunek, rok</b>	Informatyka, II rok, st. stacjonarne (3,5-l)
<b>Temat projektu</b>	<i>Platforma wysyłkowego wypożyczania filmów Blu-Ray</i>

## Spis treści

Wstęp .....	3
Narzędzia i technologie .....	4
Język programowania:.....	4
Framework: .....	4
Baza danych:.....	4
Front-end:.....	5
Środowisko programistyczne: .....	5
Baza danych.....	6
Schemat ERD bazy danych.....	6
Opis bazy danych:.....	6
Tabela films:.....	7
Tabela items:.....	7
Tabela films_types:.....	7
Tabela addresses:.....	7
Tabela transactions: .....	8
Tabela users:.....	8
Konfiguracja połączenia z bazą danych .....	8
Relacje w modelach.....	9
Migracje.....	10
Seeder.....	11
<b>UsersTableSeeder</b> .....	11
<b>FilmTypeTableSeeder</b> .....	11
<b>FilmsTableSeeder</b> .....	11
<b>TransactionSeeder</b> .....	11

<b>AdressesTableSeeder .....</b>	<b>11</b>
Projekt GUI .....	12
Uruchamianie Aplikacji.....	15
Funkcjonalność Aplikacji.....	16
Logowanie .....	16
Rejestracja .....	17
CRUD administratora.....	17
Tworzenie nowego obiektu Film .....	17
Aktualizacja obiektu Film.....	18
Przeglądanie istniejących obiektów Film.....	18
Usuwanie obiektu Film .....	19
Przywracanie obiektu Film .....	19
Wyszukiwanie obiektu Film .....	20
Zarządzanie użytkownikami przez administratora .....	20
Przeglądanie ogólnodostępnych zasobów .....	22
Zarządzania swoimi zasobami przez użytkownika .....	24
Zarządzania swoimi danymi przez użytkownika.....	26
Przeglądanie transakcji.....	26
Zmiana danych .....	26
Dodanie adresu .....	27
Zarządzanie adresami .....	27
Edycja adresu.....	28
Usuwanie adresu .....	28
Przewracanie Adresu .....	29
Proces rejestracji nowego użytkownika aplikacji, oraz logowania się obecnych użytkowników aplikacji.....	29
Rejestracja nowego użytkownika .....	29
Logowanie istniejącego użytkownika .....	31
Logika biznesowa .....	32
Walidacja danych .....	33
Walidacja danych po stronie frontendu .....	33
Walidacja danych po stronie backendu .....	34

# Wstęp

Aplikacja do wysyłkowego wypożyczania filmów Blu-ray przynosi szereg korzyści zarówno użytkownikom, jak i administratorowi. Oto niektóre z nich:

Korzyści dla użytkowników:

1. Wygoda i oszczędność czasu: Dzięki aplikacji użytkownicy mogą składać zamówienia online, unikając konieczności stania w długich kolejkach. Wystarczy kilka kliknięć, aby wypożyczyć interesujący film, a następnie zostanie on dostarczony pod wskazany adres.
2. Duży wybór filmów: Aplikacja umożliwia dostęp do szerokiej gamy filmów Blu-ray, co daje użytkownikom możliwość wyboru spośród różnorodnych tytułów i gatunków. Informacje o filmach, takie jak opisy, oceny czy zwiastuny, pomagają w podjęciu decyzji.
3. Promocje i zniżki: Użytkownicy mają dostęp do aktualnych promocji i zniżek oferowanych przez wypożyczalnię. Mogą korzystać z różnego rodzaju rabatów, które pomagają zaoszczędzić na kosztach wypożyczenia filmów.
4. Zarządzanie kontem i adresami dostawy: Dzięki aplikacji użytkownicy mają pełną kontrolę nad swoimi kontami. Mogą aktualizować dane osobowe, zmieniać adresy dostawy oraz śledzić historię swoich wypożyczeń.

Korzyści dla administratora:

1. Efektywne zarządzanie filmami i usługami: Administrator ma możliwość dodawania, usuwania i aktualizowania dostępnych filmów wraz z ich cenami i kategoriami. Może również tworzyć atrakcyjne promocje i zniżki, przyciągając tym samym większą liczbę użytkowników.
2. Monitorowanie aktywności użytkowników: Administrator ma dostęp do danych użytkowników, takich jak historia wypożyczeń, co pozwala na lepsze zrozumienie preferencji i zachowań klientów. Dzięki temu może podejmować lepiej pomyślane decyzje dotyczące asortymentu filmów i strategii marketingowych.
3. Skuteczna kontrola nad działalnością wypożyczalni: Aplikacja zapewnia administratorowi narzędzia umożliwiające śledzenie transakcji, analizę danych. To pozwala na efektywne monitorowanie działalności wypożyczalni i podejmowanie odpowiednich działań w celu optymalizacji procesów.

Podsumowując, aplikacja wypożyczalni filmów Blu-ray przynosi liczne korzyści zarówno dla użytkowników, którzy cieszą się wygodnym procesem wypożyczania i dostępem do szerokiej gamy filmów, jak i dla administratora, który ma możliwość efektywnego zarządzania asortymentem, promocjami i danymi użytkowników.

# Narzędzia i technologie

Podczas realizacji projektu zastosowano różnorodne narzędzia, technologie, języki programowania, bazę danych, biblioteki i frameworki. Oto lista użytych komponentów:

## Język programowania:

- PHP (wersja 8.2.0): Jest to język skryptowy wykorzystywany do tworzenia aplikacji webowych. PHP umożliwia tworzenie dynamicznych stron internetowych oraz interakcję z bazami danych. Jest powszechnie stosowany w połączeniu z serwerem WWW, takim jak Apache, oraz bazami danych, takimi jak MySQL. PHP jest łatwy do nauki i posiada bogatą dokumentację, która jest dostępna na stronie php.net.

## Framework:

- Laravel (wersja 10.0): Jest to framework PHP, który zapewnia kompletny zestaw narzędzi i strukturę do szybkiego tworzenia zaawansowanych aplikacji webowych. Laravel został zaprojektowany w celu zwiększenia produktywności programistów poprzez elegancką składnię i intuicyjne rozwiązania. Można pobrać najnowszą wersję Laravel ze strony laravel.com. Framework ten jest udostępniany na zasadach licencji MIT, co oznacza, że jest darmowy w użyciu, a także posiada otwarty kod źródłowy, co umożliwia programistom dostosowanie go do swoich potrzeb. Laravel cieszy się dużą popularnością w społeczności programistów PHP ze względu na swoją rozbudowaną dokumentację, szerokie wsparcie społecznościowe i bogaty ekosystem narzędzi i bibliotek.

## Baza danych:

- MySQL: Jest to popularna relacyjna baza danych wykorzystywana do przechowywania i zarządzania różnorodnymi informacjami, takimi jak dane użytkowników, treści aplikacji, statystyki itp. MySQL zapewnia niezawodne i wydajne rozwiązania bazodanowe, które są szeroko stosowane w branży IT. Można pobrać najnowszą wersję MySQL ze strony mysql.com. MySQL oferuje różne zasady licencjonowania, w tym darmową wersję Community Edition, która jest popularna wśród programistów i małych przedsiębiorstw. Dostępność różnych wersji licencyjnych daje elastyczność w wyborze i dopasowywaniu MySQL do konkretnych potrzeb. Baza danych MySQL jest również dobrze udokumentowana, posiada rozbudowaną społeczność użytkowników i bogaty ekosystem narzędzi, co ułatwia pracę i rozwiązywanie problemów związanych z bazami danych.

## Front-end:

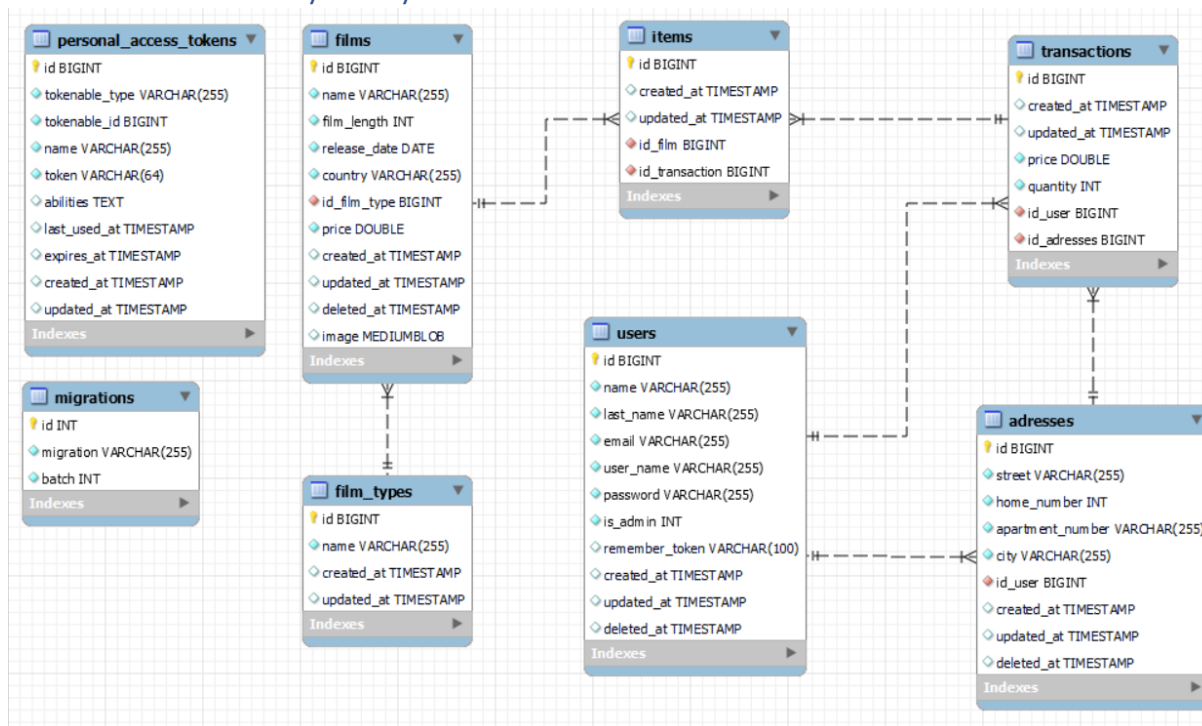
- Bootstrap (wersja 5.2.3): Bootstrap to popularny framework front-endowy, który zapewnia narzędzia, komponenty i style ułatwiające tworzenie responsywnych stron internetowych. Wersja 5.2.3 jest jedną z najnowszych wersji tego frameworka i wprowadza wiele ulepszeń i nowych funkcji.

## Środowisko programistyczne:

- Visual Studio Code: Jest to popularne środowisko programistyczne (IDE) zaprojektowane do tworzenia różnorodnych aplikacji webowych. Visual Studio Code, często nazywane VS Code, zostało stworzone przez firmę Microsoft i zdobyło szerokie uznanie wśród programistów. Można bezpłatnie pobrać najnowszą wersję Visual Studio Code ze strony [code.visualstudio.com](https://code.visualstudio.com). Dzięki swojej modularnej i elastycznej architekturze, VS Code oferuje wsparcie dla wielu języków programowania, w tym HTML, CSS, JavaScript, PHP, Python i wiele innych. Wyposażone w wiele przydatnych funkcji, takich jak automatyczne uzupełnianie kodu, debugowanie, integracja z systemem kontroli wersji, narzędzia do pracy z rozszerzeniami i wiele innych, Visual Studio Code zapewnia wydajne i wygodne środowisko do tworzenia aplikacji webowych. Bez względu na to, czy jesteś doświadczonym programistą czy dopiero zaczynasz swoją przygodę z programowaniem, Visual Studio Code jest przyjazne dla użytkownika, łatwe w użyciu i oferuje wiele możliwości dostosowania, aby dostosować je do swoich preferencji i stylu pracy.

# Baza danych

## Schemat ERD bazy danych



## Opis bazy danych:

Baza danych została stworzona w celu obsługi wysyłkowego wypożyczania filmów w formacie Blu-ray. Jej głównym celem jest gromadzenie informacji dotyczących transakcji, które są podstawą do wystawiania paragonów oraz zawierają szczegółowe dane o zakupach.

Tabela "transactions" stanowi centrum bazy danych, przechowując informacje dotyczące poszczególnych transakcji. Każda z nich zawiera datę zakupu, datę wygaśnięcia, identyfikator użytkownika, identyfikator dostawczego adresu oraz pełny koszt transakcji. Jest to podstawowe źródło informacji na temat zamówień dokonywanych przez klientów.

W tabeli "users" znajdują się dane dotyczące użytkowników, którzy dokonują zakupów. Zawiera informacje takie jak identyfikator użytkownika, imię, nazwisko, adres e-mail, nazwa użytkownika, hasło oraz rola użytkownika. Te dane są przydatne do śledzenia tożsamości osób dokonujących konkretnych transakcji.

Tabela "items" przechowuje informacje o filmach, które są zamawiane w ramach poszczególnych transakcji. Przedmioty w tej tabeli reprezentują filmy, a konkretne informacje o nich są pobierane z tabeli "films".

Dzięki powiązaniom między tymi tabelami, baza danych umożliwia wszechstronne śledzenie zamówień, identyfikację użytkowników oraz zakupionych przedmiotów (filmów). Taka struktura danych zapewnia spójność informacji oraz umożliwia analizę transakcji oraz zarządzanie zamówieniami.

Baza danych zawiera kilka tabel, takich jak: `films`, `items`, `migrations`, `personal_access_tokens`, `films_types`, `addresses`, `transactions` oraz `users`

#### Tabela `films`:

- o `id` - unikalny identyfikator filmu
- o `name` - nazwa filmu
- o `film_length` - długość trwania filmu
- o `release_date` - data wydania filmu
- o `country` - kraj wyprodukowania filmu
- o `id_film_type` - identyfikator typu filmu (klucz obcy odnosi się do tabeli `film_types`)
- o `price` - cena za wypożyczenie (za jeden dzień)
- o `created_at` - data i czas utworzenia filmu
- o `updated_at` - data i czas ostatniej aktualizacji filmu

#### Tabela `items`:

- o `id` - unikalny identyfikator pozycji
- o `id_film` - identyfikator filmu (klucz obcy odnosi się do tabeli `films`)
- o `id_transaction` - identyfikator transakcji (klucz obcy odnosi się do tabeli `transactions`)
- o `created_at` - data i czas utworzenia
- o `updated_at` - data i czas ostatniej aktualizacji

#### Tabela `films_types`:

- o `id` - unikalny identyfikator typu filmu
- o `name` - nazwa typu filmu

#### Tabela `addresses`:

- o `id` - unikalny identyfikator adresu
- o `street` - nazwa ulicy
- o `home_number` - numer domu
- o `apartment_number` - numer mieszkania (opcjonalny)
- o `city` - nazwa miasta
- o `id_user` - identyfikator użytkownika (klucz obcy odnosi się do tabeli `users`)
- o `created_at` - data i czas utworzenia adresu
- o `updated_at` - data i czas ostatniej aktualizacji adresu
- o `deleted_at` - data i czas (soft) usunięcia adresu

#### Tabela transactions:

- o `id` - unikalny identyfikator transakcji
- o `id_user` - identyfikator użytkownika (klucz obcy odnosi się do tabeli `users`)
- o `id_adresses` - identyfikator adresu (klucz obcy odnosi się do tabeli `adresses`)
- o `price` - kwota transakcji
- o `quantity` – liczba dni na jakie film został wypożyczony
- o `created_at` - data i czas utworzenia transakcji
- o `updated_at` - data i czas ostatniej aktualizacji transakcji

#### Tabela users:

- o `id` - unikalny identyfikator użytkownika
- o `name` - imię użytkownika
- o `lastname` – nazwisko użytkownika
- o `password` – hasło użytkownika
- o `email` - adres e-mail użytkownika
- o `user_name` - nazwa użytkownika w aplikacji
- o `is_admin` – rola użytkownika (admin lub użytkownik)
- o `created_at` - data i czas utworzenia użytkownika
- o `updated_at` - data i czas ostatniej aktualizacji użytkownika
- o `deleted_at` – data i czas (soft) usunięcia użytkownika

## Konfiguracja połączenia z bazą danych

Połączenie z bazą danych w pliku konfiguracyjnym `.env`

`DB_CONNECTION=mysql`

`DB_HOST=localhost`

`DB_PORT=3306`

`DB_DATABASE=filmrental`

`DB_USERNAME=root`

`DB_PASSWORD=root`



## Relacje w modelach

Wszystkie powiązania między modelami oparte są na relacji One To Many. Oto przykład relacji między modelem "User" a "Transaction":

Relacja "hasMany" w modelu "User": Metoda "transactions()" definiuje relację "posiada wiele" między użytkownikiem a transakcjami. Oznacza to, że jeden użytkownik może mieć wiele transakcji. Wykorzystywane jest pole "id\_user" w tabeli "transactions" jako klucz obcy, odnoszący się do tabeli "users". Poniżej znajduje się kod tej relacji w modelu "User":

```
public function transactions()
{
    return $this->hasMany(Transaction::class, 'id_user');
}
```

Relacja "belongsTo" w modelu "Transaction": Metoda "user()" definiuje relację "należy do" między transakcją a użytkownikiem. Oznacza to, że jedna transakcja należy do jednego użytkownika. Wykorzystywane jest pole "id\_user" w tabeli "transactions" jako klucz obcy, odnoszący się do tabeli "users". Poniżej znajduje się kod tej relacji w modelu "Transaction":

```
public function user()
{
    return $this->belongsTo(User::class, 'id_user');
}
```

Dzięki tym relacjom możliwe jest ustalenie powiązań między użytkownikami a transakcjami. Relacja "hasMany" umożliwia dostęp do wszystkich transakcji danego użytkownika, podczas gdy relacja "belongsTo" umożliwia uzyskanie informacji o użytkowniku, do którego należy konkretna transakcja.

## Migracje

Migracje są używane do zarządzania strukturą bazy danych i tworzenia, zmieniania lub usuwania tabel i kolumn. Przykładowo tabela Users:

```
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('name');
            $table->string('last_name');
            $table->string('email')->unique();
            $table->string('user_name')->unique();
            $table->string('password');
            $table->integer('is_admin')->default(0);
            $table->rememberToken();
            $table->timestamps();
            $table->softDeletes();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('users');
    }
}
```

W tym przypadku migracja tworzy tabelę "users" za pomocą metody `Schema::create()`. Tabela "users" będzie zawierać kolumny takie jak:

- `id`: Autoinkrementujące pole identyfikatora.
- `name`: Pole typu string, przechowujące imię użytkownika.
- `last_name`: Pole typu string, przechowujące nazwisko użytkownika.
- `email`: Pole typu string przechowujące adres e-mail użytkownika.
- `user_name`: Pole typu, przechowujące nazwisko użytkownika.
- `password`: Pole typu string przechowujące hasło użytkownika.
- `is_admin`: Pole typu boolean przechowujące informację o roli użytkownika. Domyślnie ustawione na 0.
- `timestamps()`: Automatycznie generowane pola "created\_at" i "updated\_at", które przechowują datę i czas utworzenia i aktualizacji rekordu.
- `softDeletes()`: Pole "deleted\_at", które służy do przechowywania informacji o usunięciu rekordu (jeśli zostanie użyte narzędzie soft-delete w Laravel).

Metoda `up()` jest odpowiedzialna za wykonanie migracji, czyli utworzenie tabeli "users" z odpowiednimi kolumnami. Metoda `down()` jest odpowiedzialna za cofnięcie migracji, czyli usunięcie tabeli "users".

## Seeder

**UsersTableSeeder:** Tworzy pięciu użytkowników: 2 administratorów administratora i trzech użytkowników. Każdy użytkownik ma unikalny adres email i hasło 'password'.

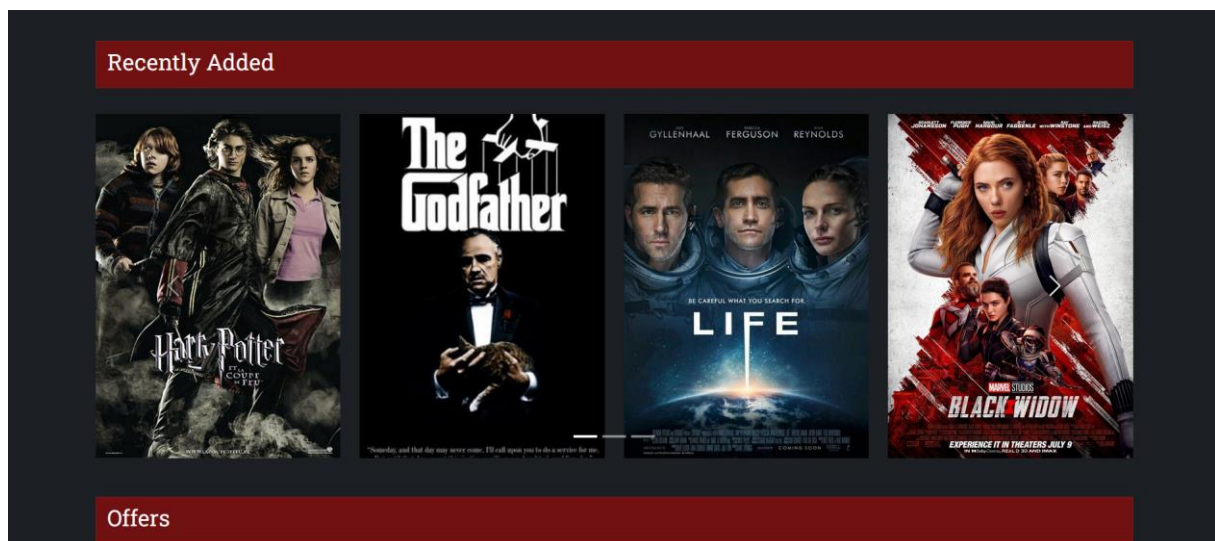
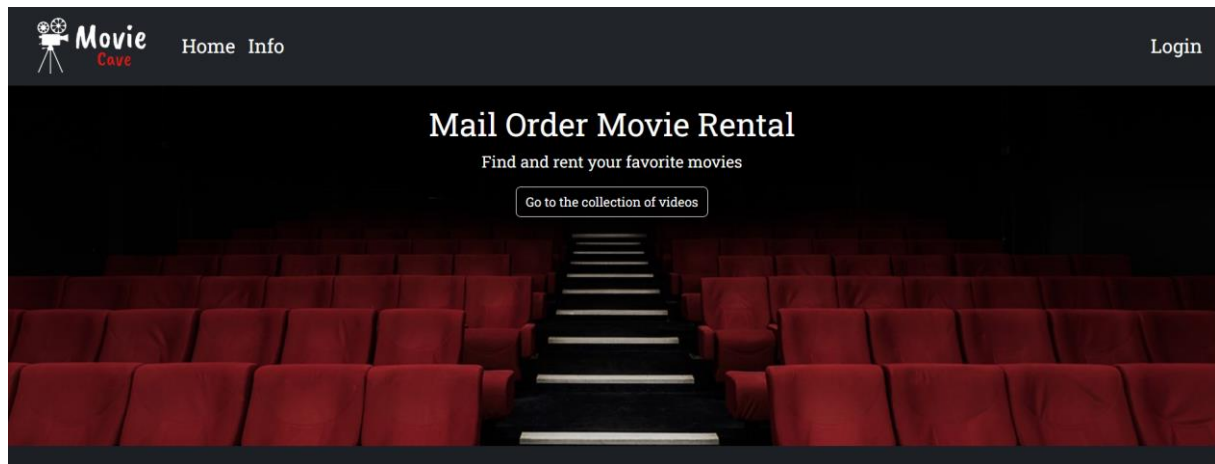
**FilmTypeTableSeeder:** Tworzy sześć różnych typów filmów: 'Comedy', 'Drama', 'Adventure', 'Action', 'Horror' i 'thriller'.

**FilmsTableSeeder:** Tworzy 32 różne filmy. Każdy film zawiera swoją nazwę, cenę za dzień wypożyczenia, nazwę kraju w którym powstał, datę wydania, id typu filmu oraz czas w minutach.

**TransactionSeeder:** Tworzy transakcje dla użytkowników nie będących administratorami. Każda transakcja zawiera datę zakupu, datę wygaśnięcia, koszt całkowity zamówienia (losowany z przedziału 8 do 20) id adresu na jaki została wykonana oraz liczbę dni na jaką było wykonane zamówienie.

**AdressesTableSeeder:** Tworzy adresy dla użytkowników nie będących administratorami.

# Projekt GUI



## Offers

Create an account now and receive a 20% discount on your first order!

Receive a 5% discount on your order if it is \$100 or more!

Offer for regular customers after the purchase of 200 films fixed 3% discounts!

## Films



Movie Rental © 2023. All rights reserved.



Movie  
Cave

[Home](#) [Info](#)

[Login](#)



### Blu-ray Promotion

Discover the world of Blu-ray with our extensive collection of high-definition movies. Immerse yourself in stunning visuals and superior audio quality.



### Blu-ray Collection

Explore our vast collection of Blu-ray movies. From action-packed blockbusters to captivating dramas, we have something for everyone.



### Upgrade to Blu-ray

Experience your favorite movies like never before. Upgrade to Blu-ray and enjoy crystal-clear picture quality and immersive surround sound.

## Experience Blu-ray. It'll blow your mind.

Indulge in the breathtaking visuals and incredible detail of Blu-ray. Immerse yourself in the world of high-definition entertainment.



## Discover Blu-ray Excellence. See for yourself.

Unleash the full potential of your home theater setup with Blu-ray. Enjoy unparalleled audiovisual quality and a truly immersive cinematic experience.

## Unleash the Power of Blu-ray. Checkmate.

Embrace the ultimate home entertainment experience with Blu-ray. Elevate your movie nights and enjoy a theater-like atmosphere in the comfort of your own home.



# Uruchamianie Aplikacji

Aby uruchomić aplikację opartą należy wykonać następujące kroki:

1. Upewnij się, że na swoim komputerze masz zainstalowane następujące narzędzia: Composer, PHP i MySQL
2. Rozpakuj archiwum projektu na swoim komputerze.
3. W pliku php.ini należy odkomentować linię extension=gd
4. Otwórz plik start.bat który zawiera następujący kod:

```
%systemDrive%\xampp\mysql\bin\mysql -uroot -e "CREATE  
DATABASE IF NOT EXISTS filmrental;"
```

```
if %errorlevel% neq 0 msg %username% "Nie udało się utworzyć  
bazy danych." && exit /b %errorlevel%
```

```
php -r "copy('.env.example', '.env');"
```

```
call composer install
```

```
call php artisan migrate:fresh --seed
```

```
call php artisan key:generate
```

```
call php artisan serve
```



Po wykonaniu powyższych kroków aplikacja powinna być dostępna pod adresem `http://localhost:8000`, gdzie będziesz mógł ją przeglądać i z niej korzystać.

Dodatkowo, przed uruchomieniem pliku `start.bat`, upewnij się, że masz działającą instancję serwera MySQL, która odpowiada konfiguracji bazy danych w pliku `.env`. Aby tego dokonać, wykonaj następujące kroki:

1. Upewnij się, że serwer MySQL jest uruchomiony na porcie 3306.
2. Utwórz nową bazę danych o nazwie "filmrental".
3. Sprawdź, czy dane uwierzytelniania do bazy danych (nazwa użytkownika i hasło) są poprawnie skonfigurowane w pliku `.env`.

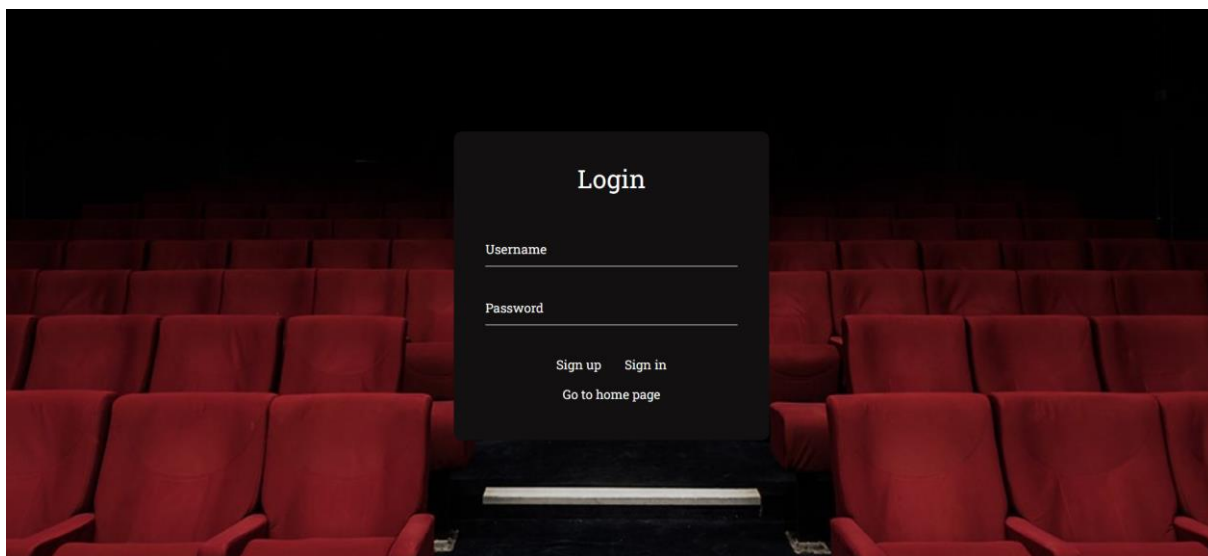
Po wykonaniu tych czynności możesz uruchomić plik `start.bat`, aby rozpocząć proces uruchamiania projektu Laravela.

## Funkcjonalność Aplikacji

### Logowanie

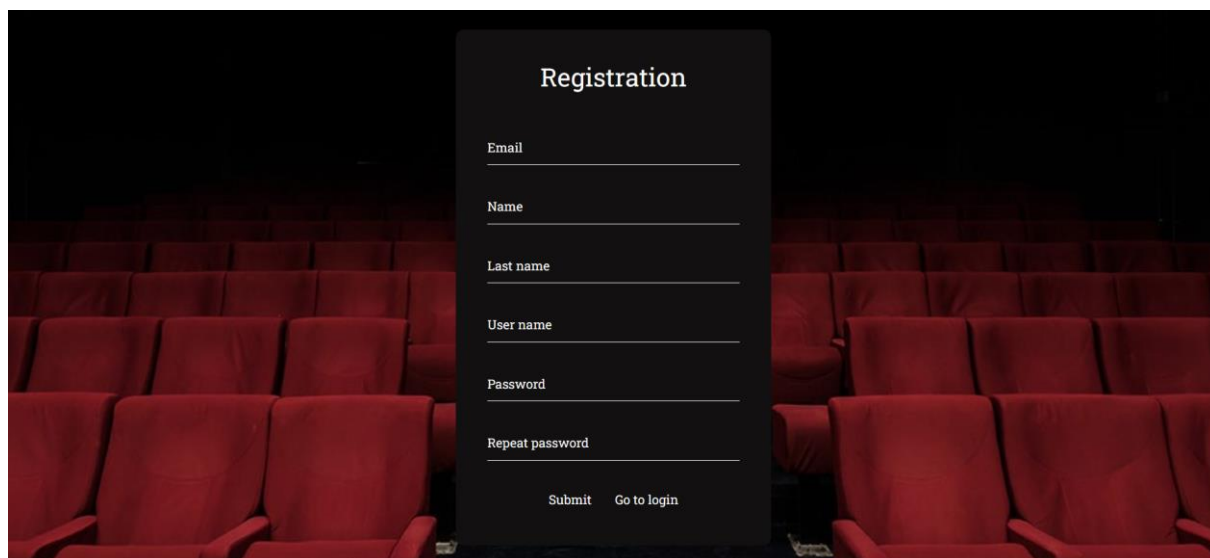
Przykładowy login i hasło jako admin: `janesmith` `password123`

Przykładowy login i hasło jako użytkownik: `emilydavis` `password123`





## Rejestracja

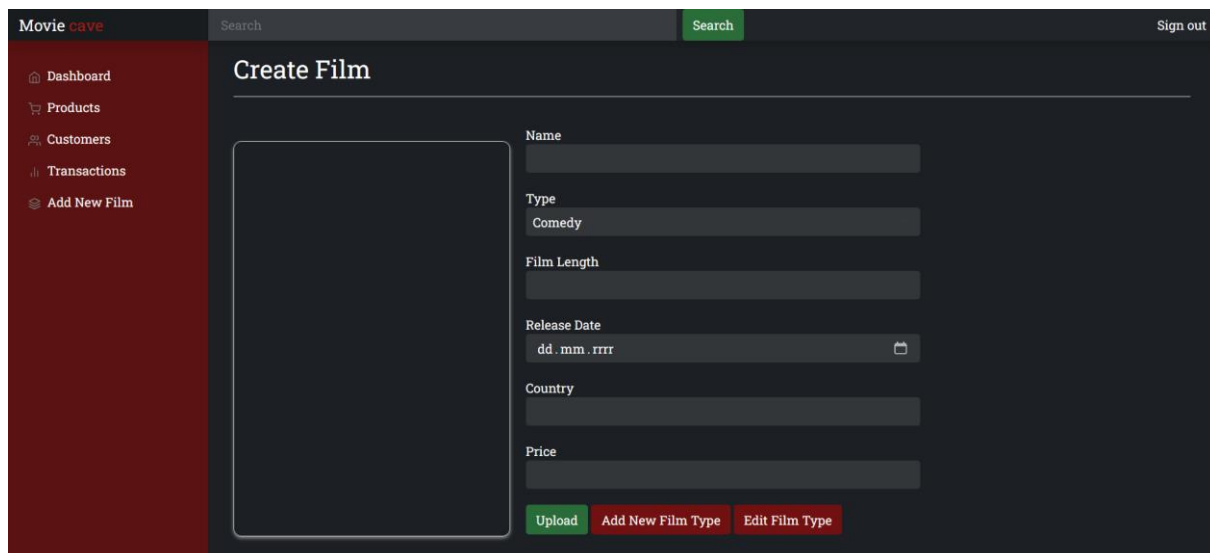


The registration form is a dark gray modal box centered over a background image of a movie theater with rows of red seats. The form contains the following fields and buttons:

- Registration** (Title)
- Email
- Name
- Last name
- User name
- Password
- Repeat password
- Submit
- Go to login

## CRUD administratora

### Tworzenie nowego obiektu Film



The 'Create Film' form is part of the MovieCave admin interface. It features a dark sidebar with navigation links and a main content area with a form and action buttons.

**MovieCave** | Search | Search | Sign out

**Create Film**

**Dashboard**  
Products  
Customers  
Transactions  
Add New Film

**Name**

**Type**  
Comedy

**Film Length**

**Release Date**  
dd . mm . yyyy

**Country**

**Price**

**Upload** **Add New Film Type** **Edit Film Type**

## Aktualizacja obiektu Film

Movie cave

Search

Search

Sign out

Dashboard


Products

Customers

Transactions

Add New Film

### Edit Film



Name

Atlantis

Type

Comedy

Film Length

120

Release Date

01.01.2022

Country

USA

Price

9.99

Upload

## Przeglądanie istniejących obiektów Film

Movie cave

Search

Search

Sign out

Dashboard





















Products

Customers

Transactions

Add New Film

### Products

#	Image	Name	Type	Film Length	Release Date	Country	Price	Actions
1		Atlantis	Comedy	120	2022-01-01	USA	\$9.99	  
2		Godzilla	Drama	90	2022-02-01	Japan	\$8.99	  
3		Hunter Killer	Adventure	120	2022-01-01	Russia	\$9.99	  
4		Journey	Action	90	2022-02-01	USA	\$8.99	  
5		Journey 2	Horror	120	2022-01-01	UK	\$9.99	  

## Usuwanie obiektu Film

The screenshot shows the 'MovieCave' dashboard with a sidebar menu on the left containing 'Dashboard', 'Products', 'Customers', 'Transactions', and 'Add New Film'. The main area is titled 'Products' and contains a table with columns: #, Image, Name, Type, Film Length, Release Date, Country, Price, and Actions. The table lists five films: Atlantis, Godzilla, Hunter Killer, Journey, and Journey 2. A confirmation dialog box is overlaid on the first row (Atlantis), displaying the text '127.0.0.1:8000 Confirm delete?' with 'OK' and 'Anuluj' buttons.

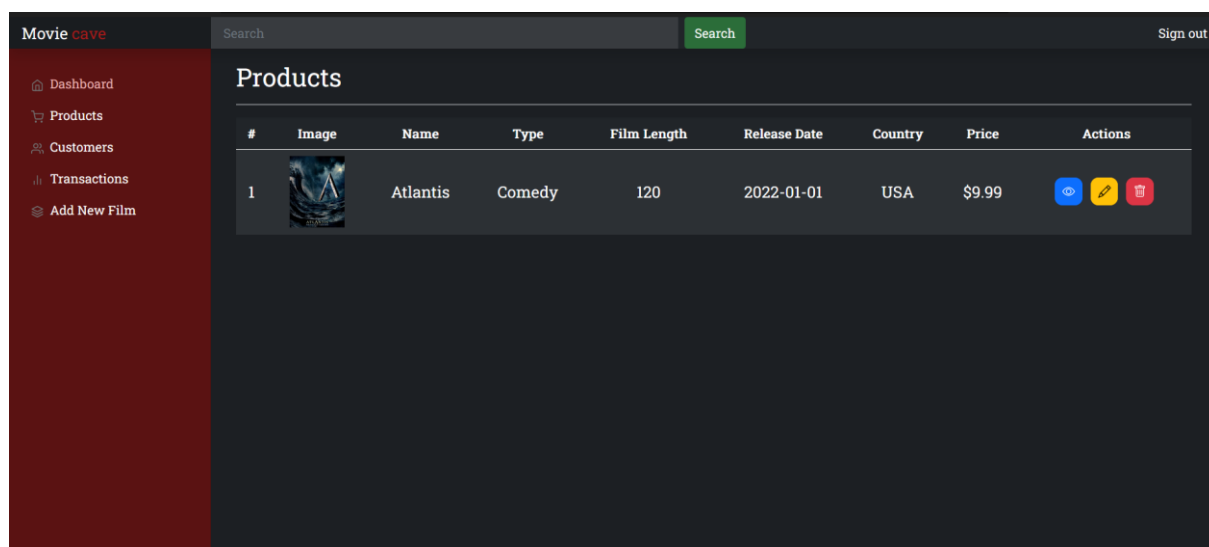
#	Image	Name	Type	Film Length	Release Date	Country	Price	Actions
1		Atlantis	Comedy	120	2022-01-01	USA	\$9.99	
2		Godzilla	Drama	90	2022-02-01	Japan	\$8.99	
3		Hunter Killer	Adventure	120	2022-01-01	Russia	\$9.99	
4		Journey	Action	90	2022-02-01	USA	\$8.99	
5		Journey 2	Horror	120	2022-01-01	UK	\$9.99	

## Przywracanie obiektu Film





The screenshot shows the 'MovieCave' dashboard after the deletion of the 'Atlantis' film. The 'Products' table now lists four films: Atlantis [DELETED], Godzilla, Hunter Killer, and Journey. The 'Atlantis [DELETED]' row has a green restore icon in the 'Actions' column. The 'Godzilla' row has been moved to the top of the list.

#	Image	Name	Type	Film Length	Release Date	Country	Price	Actions
1		Atlantis [DELETED]	Comedy	120	2022-01-01	USA	\$9.99	
2		Godzilla	Drama	90	2022-02-01	Japan	\$8.99	
3		Hunter Killer	Adventure	120	2022-01-01	Russia	\$9.99	
4		Journey	Action	90	2022-02-01	USA	\$8.99	
5		Journey 2	Horror	120	2022-01-01	UK	\$9.99	

## Wyszukiwanie obiektu Film



The screenshot displays the 'MovieCave' application interface. On the left is a dark red sidebar with navigation links: Dashboard, Products, Customers, Transactions, and Add New Film. The main area has a dark background with a 'Search' bar at the top. Below the search bar is a table titled 'Products'. The table has columns for #, Image, Name, Type, Film Length, Release Date, Country, Price, and Actions. One product is listed: 'Atlantis' (Comedy, 120 min, released 2022-01-01, USA, \$9.99). The Actions column contains three icons: a blue circle with a white 'i', a yellow pencil, and a red trash can.

#	Image	Name	Type	Film Length	Release Date	Country	Price	Actions
1		Atlantis	Comedy	120	2022-01-01	USA	\$9.99	  

## Zarządzanie użytkownikami przez administratora

W aplikacji, administrator ma możliwość efektywnego zarządzania użytkownikami poprzez przeglądanie wszystkich dostępnych użytkowników, ich transakcji oraz usuwanie lub przywracanie użytkowników, którzy zostali usunięci.

Aplikacja umożliwia administratorowi przejrzanie listy wszystkich użytkowników, co daje pełen przegląd zarejestrowanych kont. Administrator może również sprawdzić szczegóły transakcji poszczególnych użytkowników, co pozwala na śledzenie ich historii zakupów i wypożyczeń. W przypadku konieczności usunięcia użytkownika z systemu, administrator ma możliwość wykonania tej operacji. Dodatkowo, jeśli zajdzie potrzeba przywrócenia usuniętego użytkownika, administrator ma możliwość przywrócenia konta.

Dzięki temu, aplikacja zapewnia administratorowi pełną kontrolę nad zarządzaniem użytkownikami, umożliwiając wygodne przeglądanie, edycję i przywracanie kont, co przyczynia się do efektywnego zarządzania użytkownikami.

Moviecave

Search

Search

Sign out

Dashboard

Products

Customers

Transactions

Add New Film

## Users

#	Name	Last Name	Email	User Name	Is Admin	Actions
1	John	Doe	john DOE@example.com	john DOE	Yes	<div><div></div><div></div><div></div></div>
2	Jane	Smith	jane Smith@example.com	jane Smith	Yes	<div><div></div><div></div><div></div></div>
3	Robert	Johnson	robert Johnson@example.com	robert Johnson	No	<div><div></div><div></div><div></div></div>
4	Emily	Davis	emily Davis@example.com	emily Davis	No	<div><div></div><div></div><div></div></div>
5	Michael	Wilson	michael Wilson@example.com	michael Wilson	No	<div><div></div><div></div><div></div></div>

Moviecave

Search

Search

Sign out

Dashboard

Products

Customers

Transactions

Add New Film

## View User

Name : John

Last Name : Doe

Email : john DOE@example.com

User Name : john DOE

Is Admin : 1

Moviecave

Search

Search

Sign out

Dashboard

Products

Customers

Transactions

Add New Film

## Edit User

Name

John

Last Name

Doe

Email

john DOE@example.com

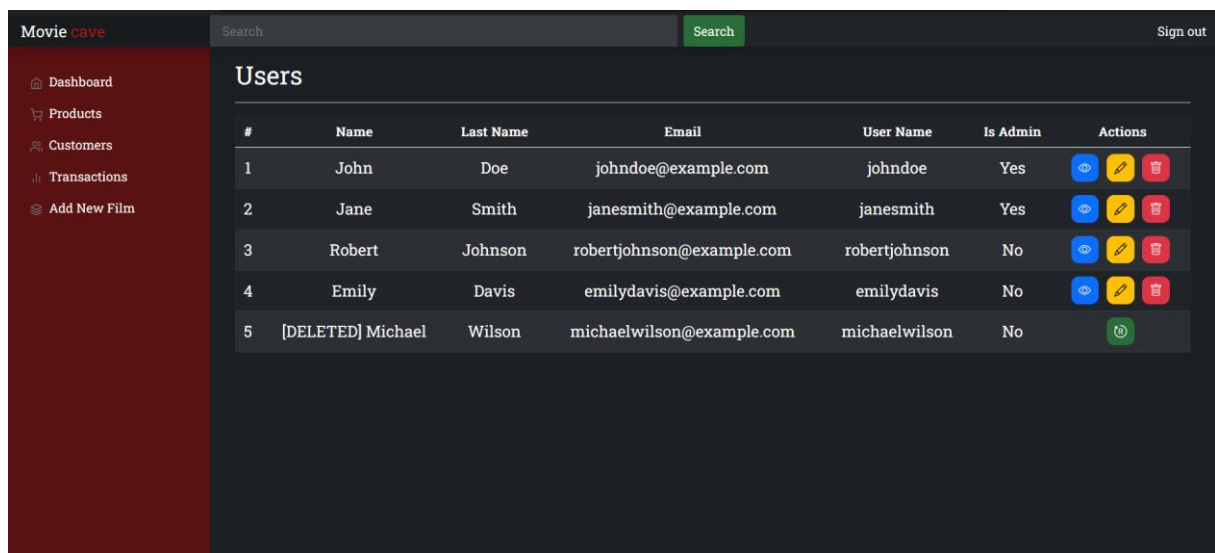
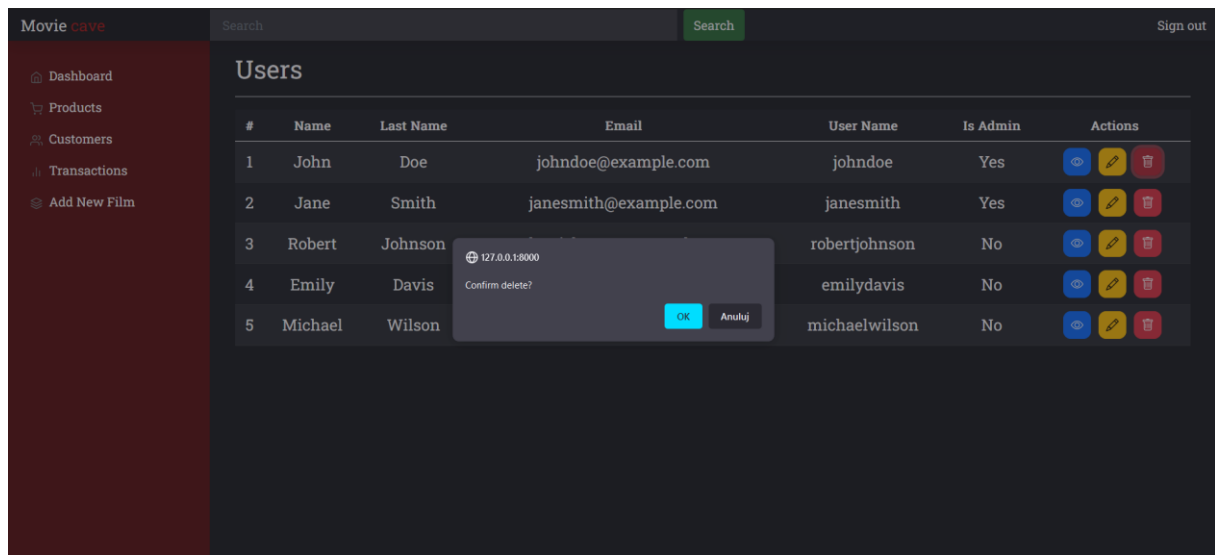
User Name

john DOE

Is Admin

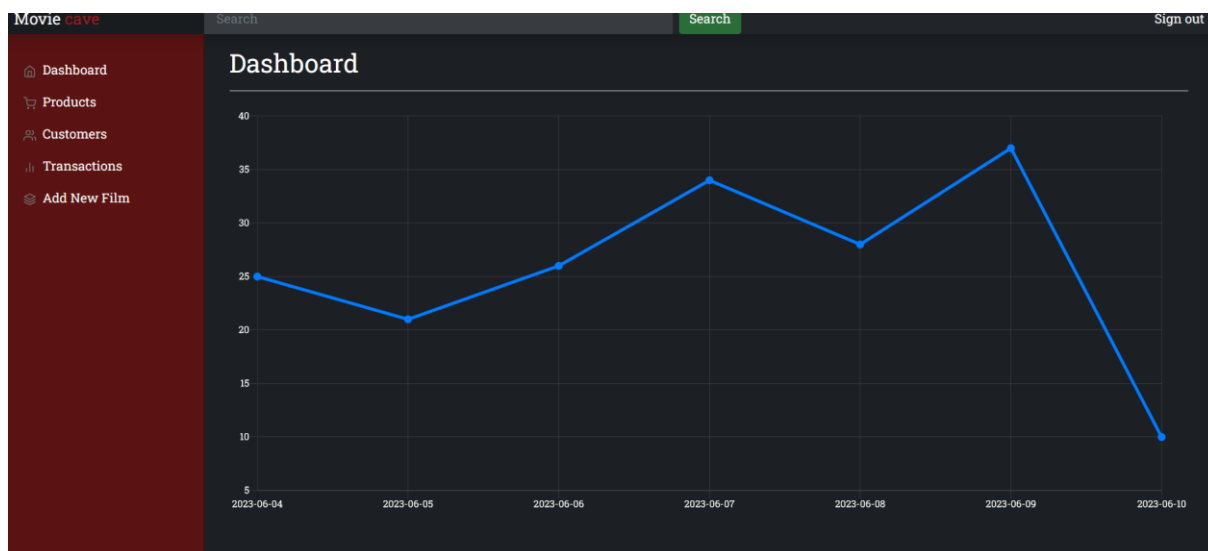
Yes

Update



## Przeglądanie ogólnodostępnych zasobów

W aplikacji administrator ma możliwość przeglądania zasobów takich jak:  
Użytkownicy, Transakcje, Filmy, Zyski z ostatniego tygodnia:



Movie cave Search Sign out

Last week's transactions

#	Date	Price	Number Of Days	User ID	Address ID
7	2023-06-04	15	5	3	1
8	2023-06-04	10	5	4	2
10	2023-06-05	11	5	3	1
11	2023-06-05	10	5	4	2
13	2023-06-06	16	5	3	1
14	2023-06-06	10	5	4	2
16	2023-06-07	15	5	3	1
17	2023-06-07	19	5	4	2
19	2023-06-08	19	5	3	1
20	2023-06-08	9	5	4	2

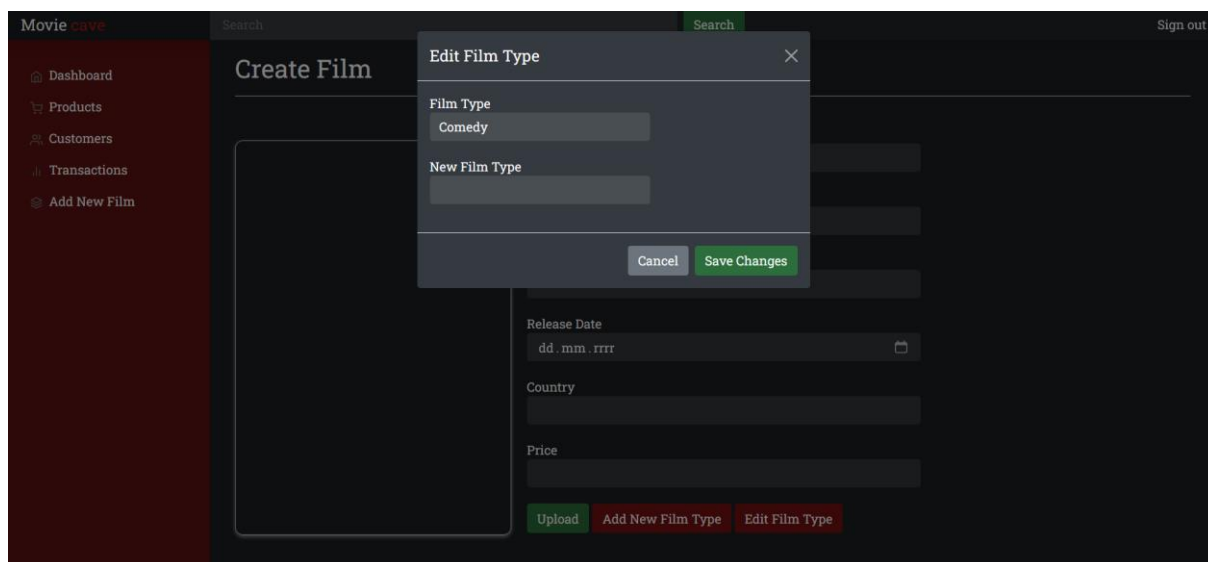
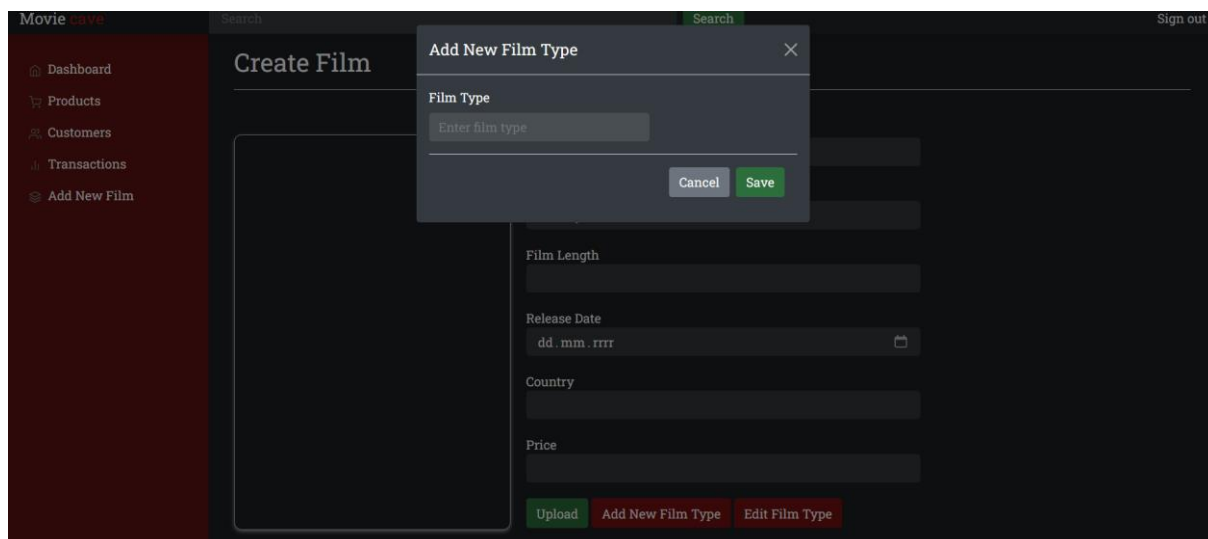
W przypadku transakcji, można jedynie przeglądać zasoby

Movie cave Search Sign out

Transactions

#	Price	Quantity	User	Address
1	\$16	5	Robert	Mathias Unions, 123, A1, Duncanberg
2	\$8	5	Emily	Shawn Manors, 456, B2, Kileymouth
3	\$15	5	Michael	Corwin Neck, 789, C3, West Santino
4	\$11	5	Robert	Mathias Unions, 123, A1, Duncanberg
5	\$20	5	Emily	Shawn Manors, 456, B2, Kileymouth
6	\$17	5	Michael	Corwin Neck, 789, C3, West Santino
7	\$14	5	Robert	Mathias Unions, 123, A1, Duncanberg
8	\$20	5	Emily	Shawn Manors, 456, B2, Kileymouth
9	\$8	5	Michael	Corwin Neck, 789, C3, West Santino

W aplikacji administrator ma możliwość dodawania nowych typów filmów oraz ich edycję:







## Zarządzania swoimi zasobami przez użytkownika









Użytkownicy mają możliwość zarządzania swoimi zasobami poprzez dokonywanie transakcji, które umożliwiają im wypożyczenie filmów.


Proces zamawiania filmu rozpoczyna się od wyboru pożądanego filmu przez użytkownika. Po dokonaniu wyboru, użytkownik przechodzi do koszyka, gdzie może zobaczyć podsumowanie swojego zamówienia. Następnie, użytkownik jest kierowany do strony płatności, gdzie może wybrać swój adres dostawy oraz preferowaną formę płatności. Po dokonaniu tych wyborów, użytkownik ma możliwość złożenia zamówienia, klikając przycisk "Pay".



Dzięki temu procesowi, użytkownicy mają pełną kontrolę nad zarządzaniem swoimi zasobami filmowymi, umożliwiając im wygodne dokonywanie transakcji i wypożyczanie filmów zgodnie z ich preferencjami.

#	Image	Name	Type	Film Length	Release Date	Country	Price	Actions
1		Atlantis	Comedy	120	2022-01-01	USA	\$7.49	<a href="#">Add to Basket</a>
2		Godzilla	Drama	90	2022-02-01	Japan	\$8.99	<a href="#">Add to Basket</a>
3		Hunter Killer	Adventure	120	2022-01-01	Russia	\$9.59	<a href="#">Add to Basket</a>
4		Journey	Action	90	2022-02-01	USA	\$8.39	<a href="#">Add to Basket</a>

       	<a href="#">Pay</a> <a href="#">Continue Shopping</a> <p>Select your address</p> <a href="#">Add address</a> <p>Select Address:</p> <div>Corwin Neck, 789, C3, West Santino</div>
---	---

#	Image	Name	Price	Number Of Days	Total Piece Prices	Actions
1		Atlantis	\$7.49	<input type="text" value="1"/> <a href="#">Update</a>	\$ 7.49	<a href="#">Remove</a>

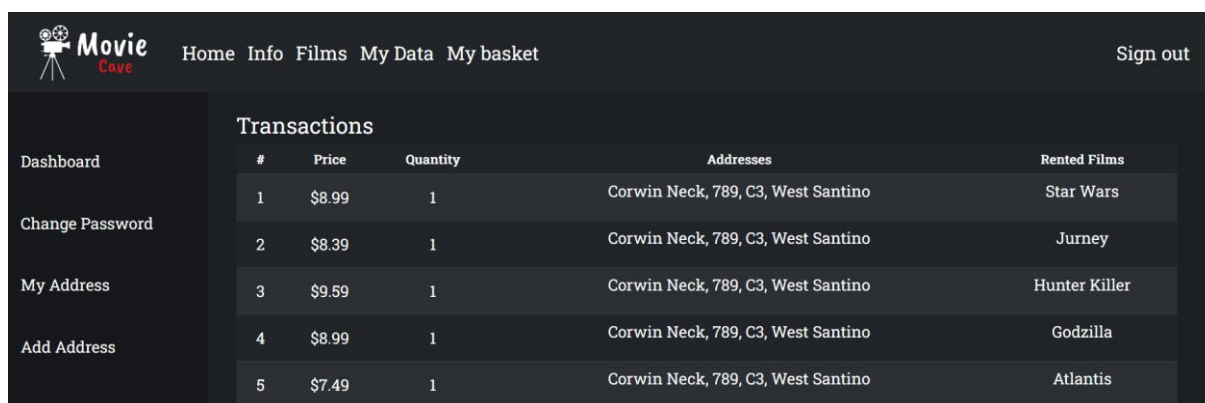
\$7.49

[Continue Shopping](#)
[Pay](#)

## Zarządzania swoimi danymi przez użytkownika

Użytkownicy mają możliwość edycji swojego konta poprzez edycję swojego imienia, nazwiska, adresu e-mail oraz hasła. Dodatkowo, użytkownicy mają dostęp do historii swoich transakcji, gdzie mogą przeglądać informacje dotyczące wcześniej dokonanych zakupów. Mogą również zarządzać swoimi adresami, dodając nowe adresy, usuwając istniejące lub aktualizując istniejące dane adresowe. Dzięki tym funkcjom, użytkownicy mają pełną kontrolę nad swoimi danymi i mogą dostosować swoje konto do swoich indywidualnych preferencji.

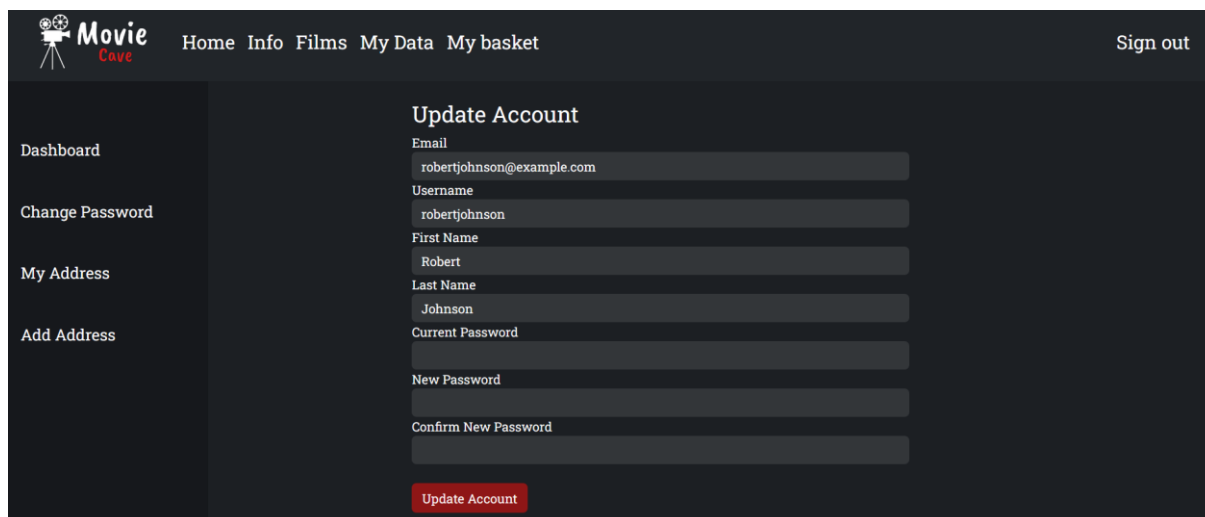
### Przeglądanie transakcji



The screenshot shows the 'Movie Cave' user dashboard. The top navigation bar includes 'Home', 'Info', 'Films', 'My Data', 'My basket', and a 'Sign out' link. The left sidebar contains links to 'Dashboard', 'Change Password', 'My Address', and 'Add Address'. The main content area is titled 'Transactions' and displays a table with the following data:

#	Price	Quantity	Addresses	Rented Films
1	\$8.99	1	Corwin Neck, 789, C3, West Santino	Star Wars
2	\$8.39	1	Corwin Neck, 789, C3, West Santino	Journey
3	\$9.59	1	Corwin Neck, 789, C3, West Santino	Hunter Killer
4	\$8.99	1	Corwin Neck, 789, C3, West Santino	Godzilla
5	\$7.49	1	Corwin Neck, 789, C3, West Santino	Atlantis

### Zmiana danych



The screenshot shows the 'Movie Cave' user dashboard with the 'Update Account' form. The top navigation bar and left sidebar are identical to the previous screenshot. The main content area is titled 'Update Account' and contains the following form fields:

- Email: robertjohnson@example.com
- Username: robertjohnson
- First Name: Robert
- Last Name: Johnson
- Current Password: [input field]
- New Password: [input field]
- Confirm New Password: [input field]

At the bottom of the form is a red button labeled 'Update Account'.

## Dodanie adresu

The screenshot shows the 'Add Address' form in the Movie Cave application. The form is located in the main content area, with a sidebar on the left containing navigation links: Dashboard, Change Password, My Address, and Add Address. The top navigation bar includes the Movie Cave logo, Home, Info, Films, My Data, My basket, and a Sign out button. The form itself has the title 'Add Address' and contains four input fields: Street, Home Number, Apartment Number, and City. A green 'Save' button is positioned below the input fields.

Movie Cave

Home Info Films My Data My basket Sign out

Dashboard

Change Password

My Address

Add Address

### Add Address

Street:

Home Number:

Apartment Number:

City:

Save

## Zarządzanie adresami

The screenshot shows the 'Addresses' table in the Movie Cave application. The table is located in the main content area, with a sidebar on the left containing navigation links: Dashboard, Change Password, My Address, and Add Address. The top navigation bar includes the Movie Cave logo, Home, Info, Films, My Data, My basket, and a Sign out button. The table has six columns: #, Street, Home Number, Apartment Number, City, and Actions. There is one row of data with the following values: 1, Corwin Neck, 789, C3, West Santino. The Actions column contains two icons: a yellow pencil icon and a red trash can icon.

Movie Cave

Home Info Films My Data My basket Sign out



Dashboard

Change Password

My Address

Add Address

### Addresses

#	Street	Home Number	Apartment Number	City	Actions
1	Corwin Neck	789	C3	West Santino	 

## Edycja adresu



The screenshot shows the 'Edit Address' form in the Movie Cave application. The form is located in the main content area, with a sidebar on the left containing navigation links: Dashboard, Change Password, My Address, and Add Address. The top navigation bar includes Home, Info, Films, My Data, My basket, and a Sign out button. The form fields are as follows:

Field	Value
Street:	Corwin Neck
Home Number:	789
Apartment Number:	C3
City:	West Santino

A green 'Save' button is located at the bottom of the form.

## Usuwanie adresu

The screenshot shows the 'Addresses' table in the Movie Cave application. The table has columns for #, Street, Home Number, Apartment Number, City, and Actions. A confirmation dialog is displayed over the table, asking to confirm the deletion of the first address.

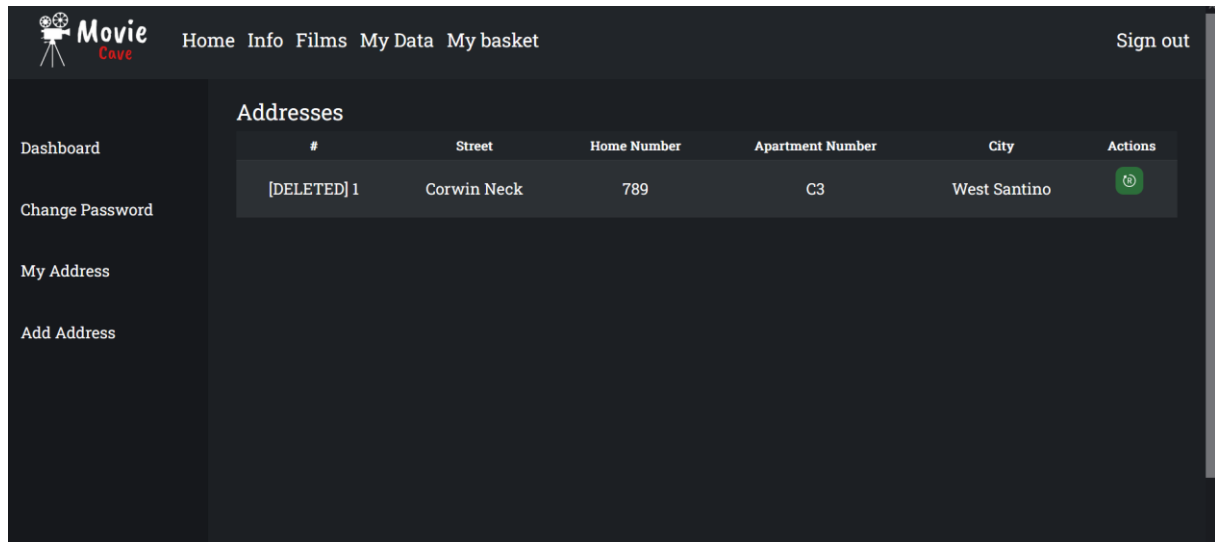
#	Street	Home Number	Apartment Number	City	Actions
1	Corwin Neck	789	C3	West Santino	 

Confirmation dialog:

127.0.0.1:8000  
Confirm delete?

Buttons: OK, Anuluj

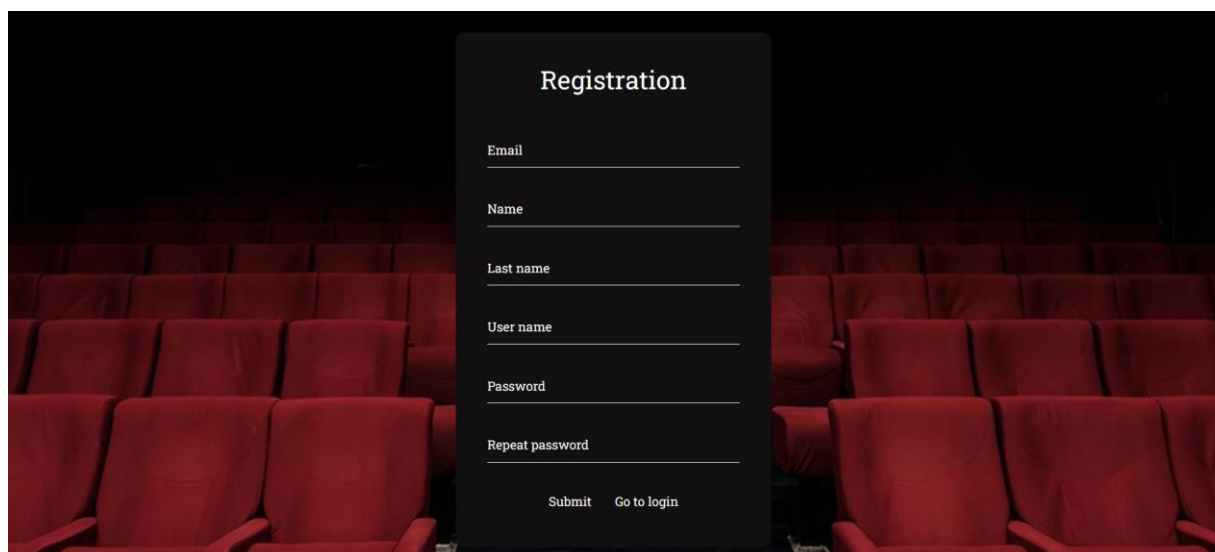
## Przewracanie Adresu



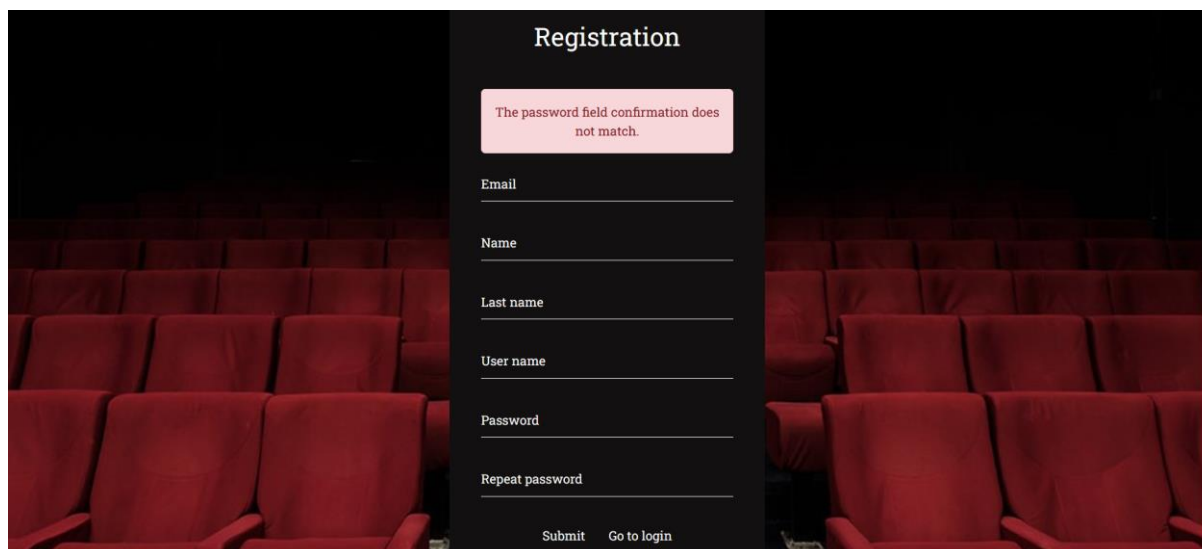
Proces rejestracji nowego użytkownika aplikacji, oraz logowania się obecnych użytkowników aplikacji

Rejestracja nowego użytkownika

Rejestracja nowego użytkownika odbywa się poprzez podstronę rejestracji

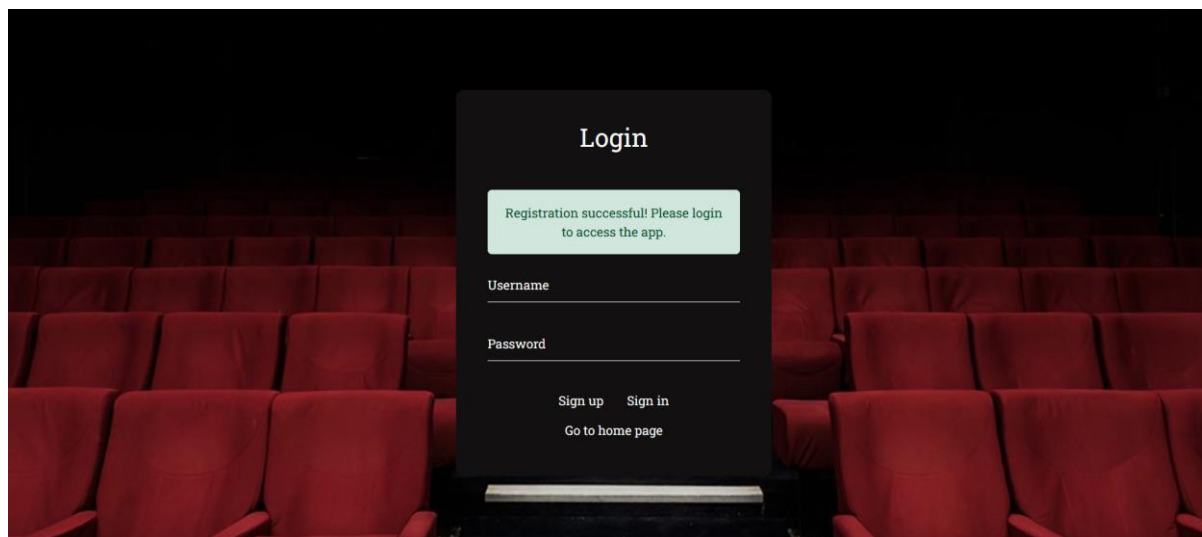


Aby zarejestrować się w systemie, użytkownik musi wprowadzić swoje dane osobowe, takie jak imię, nazwisko, nazwę użytkownika, adres e-mail, hasło oraz potwierdzenie hasła. System przeprowadza weryfikację podanych danych i w przypadku wykrycia błędów informuje użytkownika o konkretnych problemach, które należy poprawić. Dzięki takiemu podejściu użytkownik jest świadomy, jakie dane są wymagane i jakie błędy należy naprawić, aby zakończyć proces rejestracji pomyślnie.



The image shows a registration form overlay on a background of red theater seats. The form is titled "Registration" and contains a pink error message: "The password field confirmation does not match." Below the message are input fields for "Email", "Name", "Last name", "User name", "Password", and "Repeat password". At the bottom of the form are two buttons: "Submit" and "Go to login".

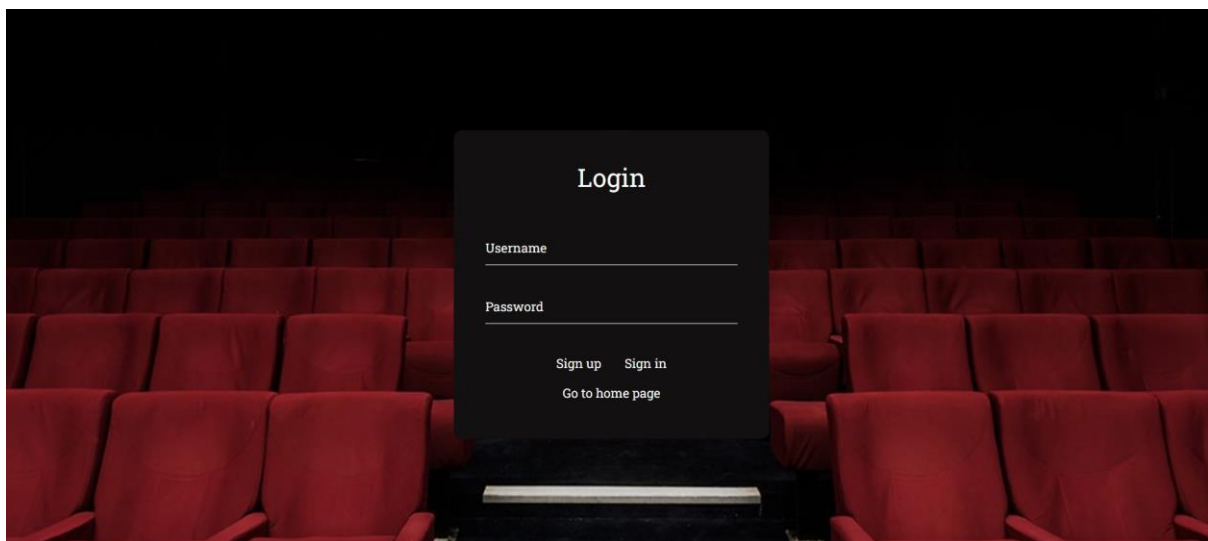
Po udanej rejestracji, użytkownik zostaje automatycznie przekierowany na stronę logowania gdzie dalej może się zalogować do aplikacji.



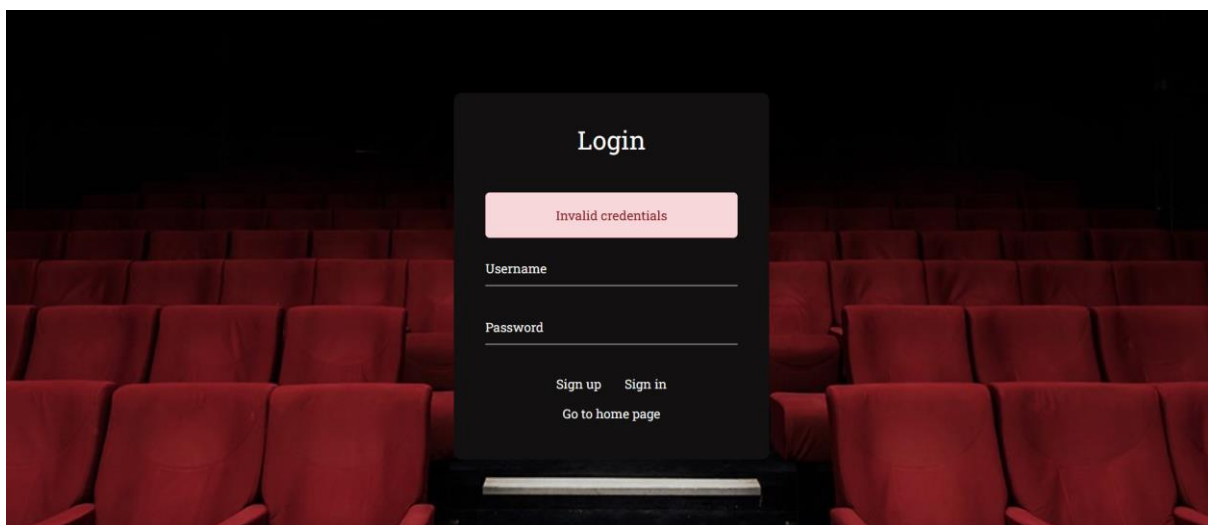
The image shows a login form overlay on a background of red theater seats. The form is titled "Login" and contains a green success message: "Registration successfull Please login to access the app." Below the message are input fields for "Username" and "Password". At the bottom of the form are three buttons: "Sign up", "Sign in", and "Go to home page".

## Logowanie istniejącego użytkownika

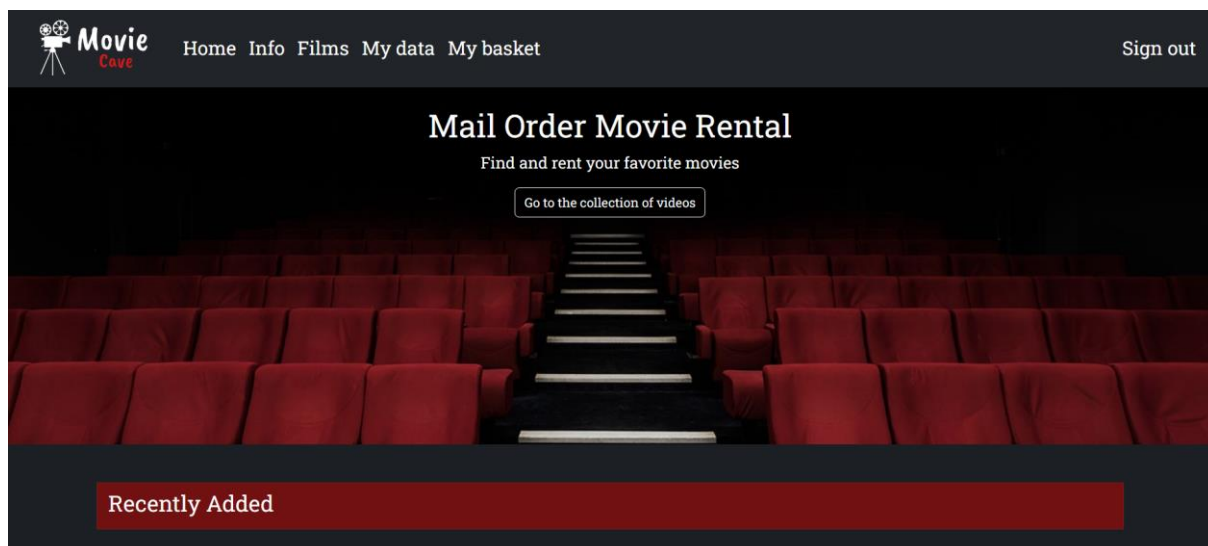
Logowanie użytkownika odbywa się poprzez podstronę logowania



W przypadku podania nie właściwych danych wyświetlony zostanie odpowiedni komunikat



Po udanym logowaniu, użytkownik zostaje przekierowany na główną stronę aplikacji, gdzie zostaje zaktualizowany pasek nawigacyjny. Teraz zamiast opcji logowania, użytkownik ma dostęp do przycisku wylogowania, który umożliwia mu bezpieczne opuszczenie konta. Dodatkowo, zalogowany użytkownik ma teraz możliwość przechodzenia na dedykowane podstrony, gdzie może składać zamówienia na filmy oraz zarządzać swoim kontem, modyfikując swoje dane personalne, zmieniając preferencje lub dodając nowe adresy dostawy. To pozwala użytkownikowi na wygodne korzystanie z pełnej funkcjonalności aplikacji i dostęp do różnorodnych opcji dostępnych dla zalogowanych użytkowników.

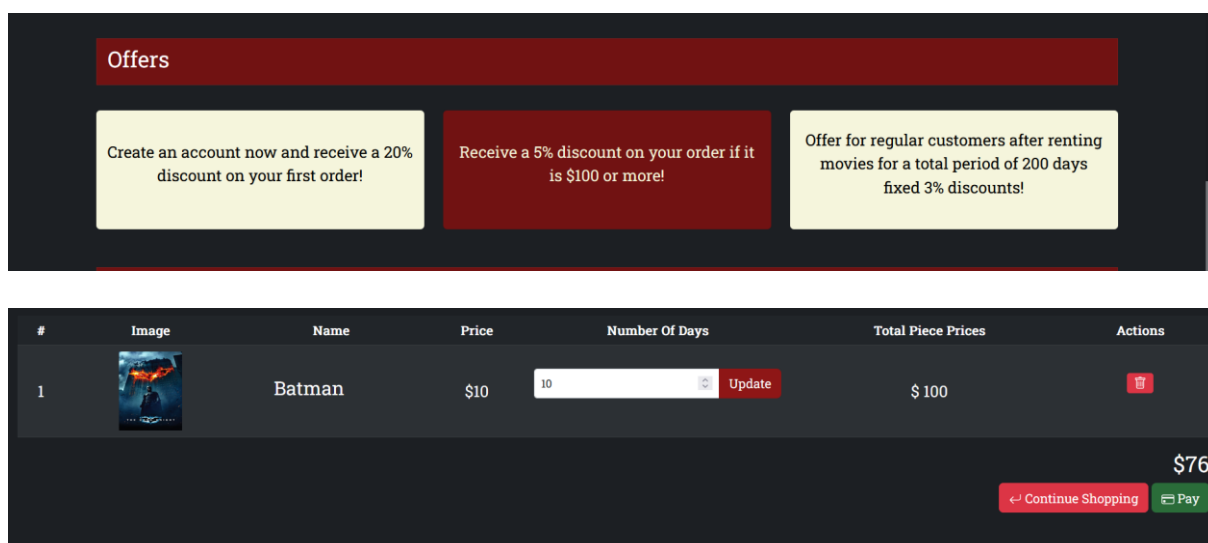


## Logika biznesowa

W aplikacji wprowadzono strategię promocyjną, która obejmuje zamówienia filmów. Jej działanie polega na przyznawaniu użytkownikowi określonych rabatów w zależności od spełnienia określonych warunków.

Po zarejestrowaniu się w aplikacji, użytkownik otrzymuje 20% rabatu na swoje pierwsze zamówienie. Ponadto, jeśli wartość zamówienia przekracza 100 dolarów, użytkownik otrzymuje 5% zniżki na całe zamówienie. Dodatkowo, każde kolejne zamówienie przynosi użytkownikowi 3% rabatu, jeśli suma dni, przez które użytkownik dokonywał zamówień filmów na swoim koncie, osiągnie 200 dni.

Ta elastyczna polityka promocyjna ma na celu zachęcenie użytkowników do regularnego korzystania z aplikacji oraz nagradzanie ich za długotrwałą aktywność. Dzięki temu użytkownicy mogą cieszyć się zniżkami na zamówione filmy, co dodatkowo zwiększa atrakcyjność korzystania z aplikacji.





## Walidacja danych

### Walidacja danych po stronie frontendu

```
<div class="user-box">
  <input type="email" name="email" required>
  <label>Email</label>
</div>
<div class="user-box">
  <input type="text" name="name" required pattern="[a-zA-Z\s]+" title="Name must contain only letters">
  <label>Name</label>
</div>
<div class="user-box">
  <input type="text" name="last_name" required pattern="[a-zA-Z\s]+" title="Last name must contain only letters">
  <label>Last name</label>
</div>
<div class="user-box">
  <input type="text" name="user_name" required minlength="5">
  <label>User name</label>
</div>
<div class="user-box">
  <input type="password" name="password" required minlength="8">
  <label>Password</label>
</div>
```

W formularzu rejestracji przedstawionym w kodzie HTML zostały zastosowane różne formy walidacji, aby upewnić się, że użytkownik wprowadza poprawne dane. Oto ogólne opisy tych form walidacji:

1. Wymagane pola:
  - W celu zapewnienia wprowadzenia niezbędnych informacji, pola zostały oznaczone jako "required". Oznacza to, że użytkownik musi wprowadzić wartości w te pola przed wysłaniem formularza.
2. Wzorce pola:
  - Aby kontrolować format danych wprowadzanych przez użytkownika, zastosowano atrybut "pattern". W przypadku pól "Name" i "Last name", wzorec "[a-zA-Z\s]+" oznacza, że pole musi zawierać tylko litery (zarówno małe, jak i wielkie) oraz spacje. To oznacza, że inne znaki lub cyfry nie będą akceptowane.
3. Minimalna długość pola:
  - Pola "User name" i "Password" mają atrybut "minlength", który określa minimalną długość wartości pola. W przypadku tych pól, minimalna długość wynosi odpowiednio 5 i 8 znaków. Oznacza to, że użytkownik musi wprowadzić co najmniej 5 znaków dla pola "User name" i co najmniej 8 znaków dla pola "Password".

Każde z tych zabezpieczeń wprowadza określone warunki, które muszą być spełnione, aby formularz został poprawnie wypełniony. Jeśli wartość wprowadzona przez użytkownika nie spełnia tych warunków, zostanie wyświetlony odpowiedni komunikat, który pomaga wskazać, jakie są oczekiwane poprawne dane.

Email

\_\_\_\_\_

Proszę wypełnić to pole

User name

asd

\_\_\_\_\_

Proszę użyć przynajmniej 5 znaków (obecnie użyto 3 znaków)

## Walidacja danych po stronie backendu

```
public function update(Request $request, string $id)
{
    $film = Film::findOrFail($id);
    $filmTypes = FilmType::pluck('id')->toArray();

    $request->validate([
        'name' => 'required',
        'id_film_type' => 'required|in:' . implode(',', $filmTypes),
        'film_length' => 'required|numeric|min:0',
        'release_date' => 'required|date|after_or_equal:' . Carbon::createFromFormat('Y-m-d', '1900-01-01')->format('Y-m-d') .
        'country' => 'required|alpha',
        'price' => 'required|numeric|min:0',
        'image' => 'required|image',
    ], [

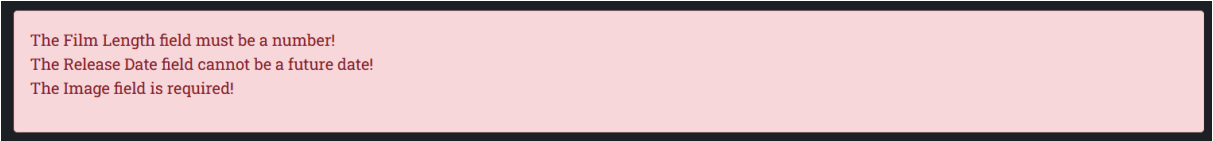
        'id_film_type.required' => 'The Type field is required!',
        'id_film_type.in' => 'The selected Type is invalid!',
        'film_length.required' => 'The Film Length field is required!',
        'film_length.numeric' => 'The Film Length field must be a number!',
        'film_length.min' => 'The Film Length field must be a non-negative number!',
        'release_date.required' => 'The Release Date field is required!',
        'release_date.date' => 'The Release Date field must be a valid date!',
        'release_date.after_or_equal' => 'The Release Date field must be greater than or equal to 1900!',
        'release_date.before_or_equal' => 'The Release Date field cannot be a future date!',
        'country.required' => 'The Country field is required!',
        'country.alpha' => 'The Country field must contain only letters!',
        'price.required' => 'The Price field is required!',
        'price.numeric' => 'The Price field must be a number!',
        'price.min' => 'The Price field must be a non-negative number!',
        'image.required' => 'The Image field is required!',
        'image.image' => 'The Image must be a valid image file!',
    ]);
}
```

Powyższy kod zawiera walidację żądań za pomocą metody `validate` dostępnej w obiekcie `Request`. Oto opis walidacji przeprowadzanej na poszczególnych polach:

1. Pole "name":
  - `required`: Wymaga wprowadzenia wartości w pole.

2. Pole "id\_film\_type":
  - required: Wymaga wprowadzenia wartości w pole.
  - in:implode(',', \$filmTypes): Wymaga, aby wartość pola znajdowała się na liście identyfikatorów dostępnych w tablicy \$filmTypes.
3. Pole "film\_length":
  - required: Wymaga wprowadzenia wartości w pole.
  - numeric: Wymaga, aby wartość pola była liczbą.
  - min:0: Wymaga, aby wartość pola była większa lub równa zero.
4. Pole "release\_date":
  - required: Wymaga wprowadzenia wartości w pole.
  - date: Wymaga, aby wartość pola była poprawną datą.
  - after\_or\_equal:1900-01-01: Wymaga, aby wartość pola była większa lub równa 1 stycznia 1900 roku.
  - before\_or\_equal:current\_date: Wymaga, aby wartość pola była mniejsza lub równa bieżącej dacie.
5. Pole "country":
  - required: Wymaga wprowadzenia wartości w pole.
  - alpha: Wymaga, aby wartość pola zawierała tylko litery.
6. Pole "price":
  - required: Wymaga wprowadzenia wartości w pole.
  - numeric: Wymaga, aby wartość pola była liczbą.
  - min:0: Wymaga, aby wartość pola była większa lub równa zero.
7. Pole "image":
  - required: Wymaga dodania wartości w pole (przesłanie pliku).
  - image: Wymaga, aby wartość pola była poprawnym plikiem obrazu.

Każde z tych warunków jest sprawdzane podczas walidacji żądania. Jeśli wartość wprowadzona przez użytkownika nie spełnia tych warunków, wyświetlane są odpowiednie komunikaty błędów, które informują o nieprawidłowych danych wprowadzonych w żądaniu.



The Film Length field must be a number!  
The Release Date field cannot be a future date!  
The Image field is required!