

Politechnika Warszawska  
Wydział Elektroniki i Technik Informacyjnych

---

**PAINT  
PROJEKT**

---

*Autorzy:*

NATALIA ŚLEPOWROŃSKA 318847  
DANIEL GÓRSKI 310855  
MIKOŁAJ RÓŻYCKI 303951  
ADRIAN OSĘDOWSKI 311104  
OLIWIA JĘDRYCZKA 318781  
KAMILA KWAŚNIEWSKA 318795

Warszawa 2023Z

## Spis treści

<b>1. Temat</b>	2
<b>2. Zespół</b>	2
<b>3. Założenia funkcjonalne aplikacji</b>	2
3.1. Przeglądanie repertuaru	2
3.2. Rezerwacja biletów	2
3.3. Kategorie filmowe	2
<b>4. Założenia architektoniczne aplikacji</b>	2
<b>5. Wykorzystanie narzędzi programowych</b>	2
<b>6. Realizacja</b>	3
6.1. Współpraca	3
6.2. Baza danych	3
6.2.1. Atrybuty	3
6.2.2. Zależności między tabelami	4
6.2.3. Przykład tworzenia tabeli	5
6.3. Front-end	6
6.4. Back-end	6
<b>7. Opis techniczny</b>	6
<b>8. Instalacja produktu</b>	7
<b>9. Opis użytkowania</b>	7

## **1. Temat**

Tematem naszego projektu było opracowanie strony internetowej kina.

## **2. Zespół**

Członkowie zespołu wraz z zakresem obowiązków:

- Oliwia Jędryczka - kierownik zespołu, programowanie front-end
- Kamila Kwaśniewska - programowanie front-end
- Mikołaj Różycki - programowanie front-end
- Adrian Osędowski - programowanie back-end
- Daniel Górski - programowanie back-end
- Natalia Ślepowrońska - programowanie bazy danych

## **3. Założenia funkcjonalne aplikacji**

### **3.1. Przeglądanie repertuaru**

W aplikacji znajdzie się repertuar filmów z aktualnymi informacjami o nich. Repertuar oprócz informacji o filmach będzie zawierać informację o kinach, salach i godzinach seansów.

Nasza strona będzie obsługiwać dwa kina w różnych lokalizacjach. Każde z nich posiada dwie sale różnej wielkości. Godziny seansów są zaplanowane na 14.00, 17.00 i 20.00 codziennie.

### **3.2. Rezerwacja biletów**

W naszej aplikacji internetowej możliwa będzie rezerwacja biletów online na wybrane seanse. Zapewniona zostanie możliwość wyboru konkretnych siedzeń. Potwierdzenia rezerwacji będą wysyłane na adres e-mail.

### **3.3. Kategorie filmowe**

Każdy film będzie opisany odpowiednimi kategoriami (gatunek, reżyser), by ułatwić wybór filmu.

## **4. Założenia architektoniczne aplikacji**

Nasz projekt obejmuje trzy obszary:

- front-end - część aplikacji, którą użytkownik odbiera i z którą bezpośrednio interakcjonuje. Obejmuje ona interfejs użytkownika, czyli to, co widzi i używa użytkownik na stronie internetowej lub w aplikacji. Wykorzystywane języki: **HTML, CSS, JavaScript**.
- back-end - część aplikacji, która działa po stronie serwera i odpowiada za przetwarzanie logiki biznesowej, komunikację z bazą danych, autoryzację użytkowników. Wykorzystywane języki: **Python, SQL**. Wybraliśmy Pythona ze względu na naszą znajomość tego języka oraz jego prostą i przejrzystą składnię ezawodni.
- baza danych - przechowuje dane uży, niwane przez aplikację. Wykorzystano relacyjną bazę danych **SQLite**. Jest ona lekka i prosta w użytkowaniu baza, dlatego postawiliśmy właśnie na nią.

## **5. Wykorzystanie narzędzi programowych**

Korzystaliśmy z różnorodnych narzędzi programowych.

Do tworzenia kodu w języku HTML korzystaliśmy ze strony replit, by na bieżąco weryfikować wygląd naszej strony.

Kod back-end w języku Python tworzyliśmy w zintegrowanych środowiskach programistycznych, takich jak min. PyCharm. Do tworzenia naszej aplikacji korzystaliśmy z framework'u Django.

Wykorzystane biblioteki:

- sqlite3
- Narzędzia do pracy zespołowej oraz komunikacji:
- Github
- Google Meet
- Lettucemeet
- Teams
- Messenger
- Discord

## 6. Realizacja

### 6.1. Współpraca

Z wykorzystaniem wyżej wspomnianych narzędzi prowadziliśmy regularne spotkania projektowe, na których podsumowywaliśmy nasze osiągnięcia i wyznaczaliśmy cele na następne dni. Praca przebiegała bez zarzutów i systematycznie.

### 6.2. Baza danych

Pierwszym krokiem było postawienie się na miejscu osoby zarządzającej kinem i zrozumienie najważniejszych kwestii organizacyjnych działania kina. Następnie skonfrontowaliśmy to z naszymi założeniami projektowymi.

W trakcie tworzenia naszej aplikacji doszliśmy do wniosku, że należy dodać tabelę "opis" by oferować więcej informacji na temat filmów. Dochodziły do tego rozszerzenia tabel o kolejne pola, by nasza strona była jak najbardziej profesjonalna i atrakcyjna dla użytkowników.

Efektem konsultacji między członkami zespołu, odpowiedzialnymi za wszystkie trzy obszary projektu jest model encji.

#### 6.2.1. Atrybuty

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
nazwa	TEXT	-
adres	TEXT	-

Tabela 1: Atrybuty encji kino

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
kino_id	INTEGER	foreign key
numer	TEXT	-
wielosc	TEXT	-

Tabela 2: Atrybuty encji sala

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
sala_id	INTEGER	foreign key
film_id	INTEGER	foreign key
data	TEXT	-
godzina	TEXT	-

Tabela 3: Atrybuty encji seans

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
opis_id	INTEGER	foreign key
nazwa	TEXT	-
kategoria	TEXT	-
kategoria2	TEXT	-
premiera	TEXT	-
czas	TEXT	-
produkcia	TEXT	-
rezyser	TEXT	-
obsada	TEXT	-
nazwa_jpg	TEXT	-

Tabela 4: Atrybuty encji film

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
opisf	TEXT	-

Tabela 5: Atrybuty encji opis

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
imie	TEXT	-
nazwisko	TEXT	-
email	TEXT	-

Tabela 6: Atrybuty encji użytkownik

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
data	TEXT	-
rzad	TEXT	-
miejsc	TEXT	-
uzytkownik_id	INTEGER	foreign key
seans_id	INTEGER	foreign key

Tabela 7: Atrybuty encji rezerwacja

Atrybut	Typ danych	Klucz
id	INTEGER	primary key
imie_i_nazwisko	TEXT	-
email	TEXT	-
numer_telefonu	TEXT	-
temat	TEXT	-
tresc	TEXT	-

Tabela 8: Atrybuty encji contact

### 6.2.2. Zależności między tabelami

Rodzaj określa kardynalność relacji między encjami.

Encje: **Kino - Sala**

Rodzaj: **1..1-1..m**

Encje: **Kino - Użytkownik**  
Rodzaj: **0..n-0..m**

Encje: **Kino - Rezerwacja**  
Rodzaj: **1..1-0..m**

Encje: **Kino - Seans**  
Rodzaj: **1..1-0..m**

Encje: **Sala - Seans**  
Rodzaj: **1..1-0..m**

Encje: **Seans - Rezerwacja**  
Rodzaj: **1..1-0..m**

Encje: **Użytkownik - Rezerwacja**  
Rodzaj: **1..1-0..m**

Encje: **Seans - Film**  
Rodzaj: **0..n-1..1**

Encje: **Film - Opis**  
Rodzaj: **1..1-1..1**

### 6.2.3. Przykład tworzenia tabeli

```
try:
    cursor.execute('''
        CREATE TABLE sala (
            id INTEGER PRIMARY KEY,
            kino_id INTEGER,
            numer TEXT,
            wielkosc TEXT,
            FOREIGN KEY (kino_id) REFERENCES kino(id)
        )
    ''')
except Exception as e:
    pass
```

Rysunek 1: Część kodu tworząca nową tablicę

```

sale = [
    (1, 1, "1", "duża"),
    (2, 1, "2", "mała"),
    (3, 2, "1", "duża"),
    (4, 2, "2", "mała")
]

try:
    cursor.executemany("INSERT INTO sala VALUES (?, ?, ?, ?)", sale)
except Exception as e:
    pass

```

Rysunek 2: Część kodu zapełniająca tablicę danymi

### 6.3. Front-end

Na początku utworzyliśmy stronę główną o nazwie "index.html". Za jej pomocą określiliśmy ogólny charakter i wygląd, które chcielibyśmy zastosować.

Głównym wyzwaniem było zastosowanie pewnych fragmentów kodu, zwłaszcza elementu "header", na różnych podstronach. Problem wynikał z faktu, że różne osoby z zespołu tworzyły różne podstrony. Po drobnych modyfikacjach udało nam się jednak dostosować nagłówek tak, aby współpracował z pozostałymi elementami strony. Musieliśmy dostosować formatowanie tego elementu poprzez określenie jego szerokości.

### 6.4. Back-end

Backend opiera się na plikach views.py, gdzie znajduje się kod każdej podstrony, pliku urls.py - ścieżki oraz models.py odpowiada za schemat bazy danych. Zadanie polegało na połączeniu w całość stworzonych plików HTML oraz ich interakcję z bazą danych.

## 7. Opis techniczny

- 1. **Wprowadzenie:**
  - Cel aplikacji
  - Zakres funkcjonalny
  - Wymagania systemowe
  - Technologie używane do budowy aplikacji
- 2. **Moduł 1: Autentykacja i Autoryzacja:**
  - Opis funkcji: Rejestracja, logowanie, resetowanie hasła
  - Bezpieczeństwo: Zastosowane protokoły, szyfrowanie danych, zarządzanie sesją
- 3. **Moduł 2: Panel użytkownika:**
  - Profil użytkownika
  - Zarządzanie danymi osobowymi
  - Historia działań użytkownika
- 4. **Moduł 3: Zarządzanie Zasobami:**
  - Dodawanie, edycja, usuwanie zasobów
  - Kategorie i tagi
- 5. **Moduł 4: Komunikacja:**
  - System powiadomień
  - Wiadomości między użytkownikami
  - Komentarze i oceny

<b>6. Moduł 5: Analiza Danych:</b>
• Raporty i statystyki
• Wizualizacje danych
• Integracja z narzędziami analitycznymi
<b>7. Moduł 6: Zarządzanie Błądami:</b>
• Logowanie błędów
• System powiadomień o awariach
• Mechanizmy odzyskiwania po błędach
<b>8. Moduł 7: Bezpieczeństwo:</b>
• Zabezpieczenia przed atakami typu SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF) itp.
• Audit bezpieczeństwa
• Kontrola dostępu
<b>9. Moduł 8: Integracje z Zewnętrznymi Systemami:</b>
• Opis integracji z usługami zewnętrznymi
• Interfejsy API
• Synchronizacja danych
<b>10. Moduł 9: Testowanie:</b>
• Plan testów
• Testy jednostkowe, testy integracyjne, testy wydajności

## 8. Instalacja produktu

Aby korzystać z naszej aplikacji, niezbędny jest komputer z zainstalowaną przeglądarką internetową. Po sklonowaniu repozytorium na którym znajduje się cały nasz projekt i otwarzeniu projektu w środowisku programistycznym, należy uruchomić kolejno komendy:

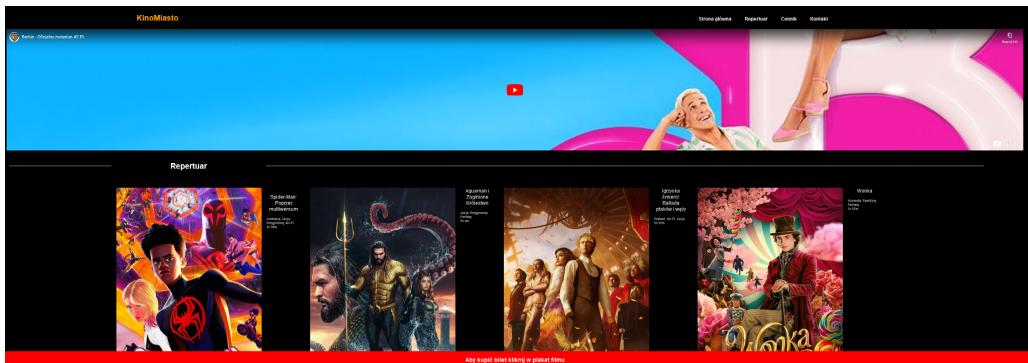
- python manage.py makemigrations
- python manage.py migrate
- python database.py
- python manage.py runserver

Pierwsze dwie komendy odpowiadają za utworzenie bazy danych i utworzeniu tabel. Następna zainicjalizację bazy danych. Kolejno cała aplikacja zostanie uruchomiona i uzyskamy możliwość sprawdzania jej funkcjonalności.

## 9. Opis użytkowania

Po otwarciu strony/projektu pierwsze, co się nam ukazuje, to strona główna kina. Strona posiada interaktywne menu (header), które przekierowuje użytkownika na kolejne strony w celu zakupu biletu kinowego, sprawdzenia aktualnego cennika lub kontaktu z kinem poprzez formularz zgłoszeniowy.

Na górze strony znajduje się odnośnik do aktualnie promowanych filmów w postaci video zaczerpniętego ze strony YouTube. Poniżej wizualizacji filmu aktualnie promowanego przechodzimy do aktualnego repertuaru kina. Dodatkowo po kliknięciu zakładki **Repertuar**, użytkownik zostaje przekierowany automatycznie do repertuaru poniżej strony głównej. Po kliknięciu odnośnika w postaci plakatu filmu w repertuarze następuje przeniesienie potencjalnego kupującego do strony filmu.



Rysunek 3: Widok strony startowej kina

Po kliknięciu użytkownik zostaje przeniesiony do strony z filmem, zawierającej krótki opis fabuły oraz kategorie fabularne, wiekowe oraz czas seansu. Poniżej opisu umieszczona jest rozwijana lista z dostępnymi kinami oraz poszczególnymi godzinami projekcji filmu, w zależności od dokonanego wyboru kina.

The screenshot shows a movie page for "Spider-Man: Across the Spider-Verse". At the top, there's a large poster for the movie. Below it, the title "Spider-Man: Poprzez multiwersum" is displayed, along with a brief plot summary: "Spider-Man: Across the Spider-Verse" (Part One) jest kontynuacją filmu z 2018 roku „Spider-Man: Uniwersum”, który cieszył się popularnością wśród na całym świecie i zarobił 375 mln dolarów. Film został nagrodzony Oscarami dla najlepszej animacji. The page also includes sections for "Informacje dodatkowe" (Additional information) and "Wybierz seans i kino" (Select a session and cinema). The "Wybierz seans i kino" section is expanded, showing a list of cinemas: "One Moony", "One Moony", "Kino Arkadia", "Kino Renbow", "Kino Targówek", and "Kino Zajazd". Below this, there's a summary of the movie's details: "Premiera: 5 stycznia 2024", "Czas trwania: 137 minut", "Produkcja: USA, 2023", "Reżyseria: Jan Kowalski", and "Oscarda: Anna Nowak, Jan Kowalski".

Rysunek 4: Widok rozwijanej listy z kinami do wyboru

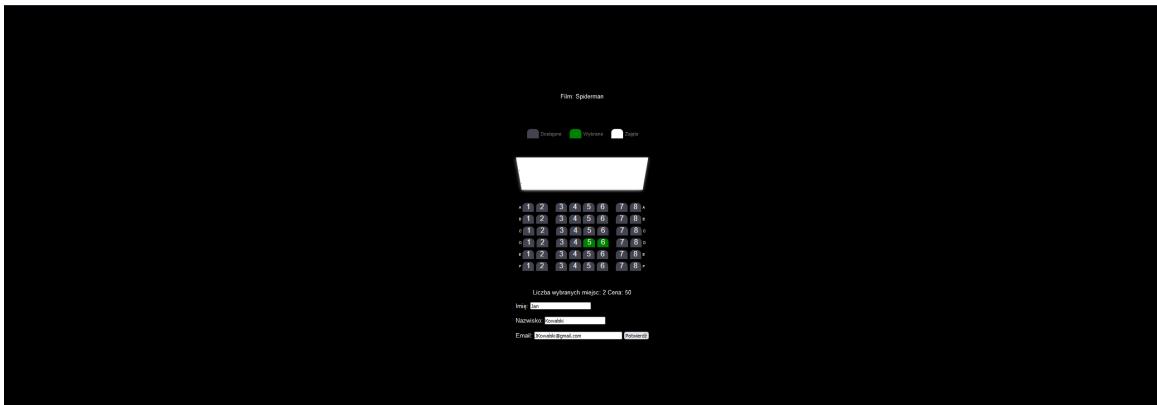
Poniżej godzin seansów umieszczona jest informacja dotycząca roku produkcji danego filmu wraz z reżyserem i obsadą. Jest to rozszerzenie opisu na górze strony.

This screenshot shows the same movie page for "Spider-Man: Across the Spider-Verse" as in Figure 4, but with the showtimes section expanded. The section is titled "Wybierz seans i kino" and contains a list of available times: "14:00 2D, PL", "17:00 2D, PL", and "20:00 2D, PL". Below the showtimes, there's a summary of the movie's details: "Premiera: 5 stycznia 2024", "Czas trwania: 137 minut", "Produkcja: USA, 2023", "Reżyseria: Jan Kowalski", and "Oscarda: Anna Nowak, Jan Kowalski".

Rysunek 5: Widok z godzinami seansów

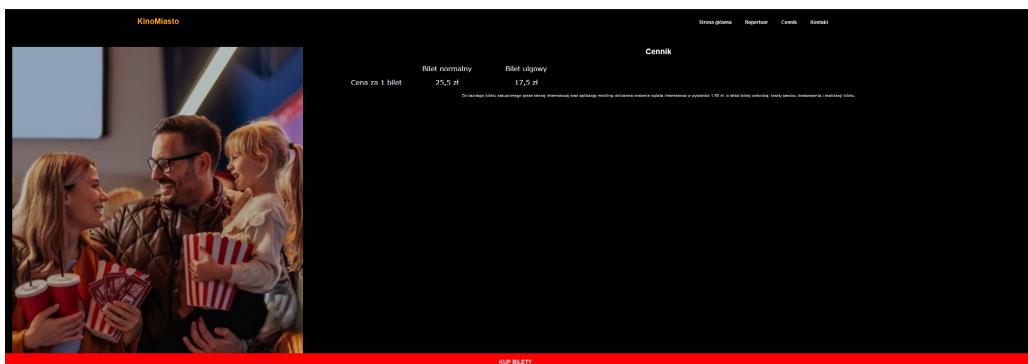
Po kliknięciu danej godziny następuje przejście do podstrony, na której użytkownik (kupujący) dokonuje zakupu biletu. Kupujący ma do wyboru wiele miejsc z sali kinowej. Po kliknięciu wybranego miejsca klient otrzymuje informację o ilości zajętych miejsc oraz potencjalnej cenie, jaką ma uściść za bilety. Zgodnie z zamieszczoną legendą, wolne miejsca są oznaczone kolorem szarym, zajęte miejsca kolorem białym, a wybrane

przez użytkownika kolorem zielonym. Po wprowadzeniu danych kontaktowych, takich jak imię, nazwisko oraz adres e-mail, i wciśnięciu przycisku '**Potwierdź**', następuje rezerwacja biletu.



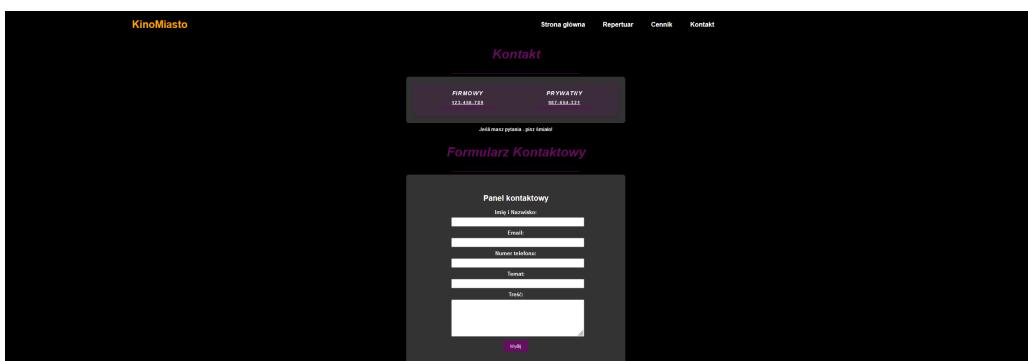
Rysunek 6: Widok sali wraz z miejscami do wyboru

Oprócz wyboru seansu i kupna biletu użytkownik ma możliwość sprawdzenia aktualnego cennika kina.



Rysunek 7: Widok cennika kina

Oprócz możliwości sprawdzenia aktualnego cennika, użytkownik ma możliwość przejścia do zakładki **Kontakt**, w której to zakładce użytkownik może się dowiedzieć, jaki jest numer kontaktowy do obsługi kina lub jaki jest numer firmowy kina. Dodatkowo, jeśli użytkownik nie ma chęci dzwonić, może złożyć zapytanie poprzez załączony formularz kontaktowy.



Rysunek 8: Widok panelu kontaktowego

The screenshot shows a MySQL query results window. The table has columns: id, imie\_i\_nazwisko, email, numer\_telefonu, temat, and tresc. There is one row with values: 1, Adrian Osędowski, adi@gazeta.pl, 787878787, Cześć, Fajne Kino.

	id	imie_i_nazwisko	email	numer_telefonu	temat	tresc
1	2	Adrian Osędowski	adi@gazeta.pl	787878787	Cześć	Fajne Kino

Rysunek 9: Widok z bazy danych z potwierdzeniem przesłania danych z formularza

Z każdej kolejnej podstrony klient ma możliwość powrotu do strony głównej po naciśnięciu żółtego logo Kina - **KinoMiasto**.