

**WYDZIAŁ TELEKOMUNIKACJI, INFORMATYKI
I ELEKTROTECHNIKI**

[Programowanie urządzeń mobilnych \[05-IST-PUM-ND1\]](#)- laboratorium

SPRAWOZDANIE Z LABORATORIUM

Lab 4 Temat sprawozdania:

Aplikacja Kotlin cz 3



**POLITECHNIKA
BYDGOSKA**

im. Jana i Jędrzeja Śniadeckich

Wykonał:

Piotrkowski Sławomir (slapio000@pbs.edu.pl)

[01-IST-PUM-ND1] rok 1 sem. 1 gr 1 stopień II

Data: 2024-12-21

1. Cel pracy

Rozbudować aplikację stworzoną w poprzednim laboratorium.

Stworzyć aplikację wyświetlającą zawartość listy z elementami String na ekranie Smartfonu z wykorzystaniem architektury Model-widok-Kontroler. Szkielet aplikacji zrealizować z użyciem komponentu Scaffold. Zaprogramować zdarzenie wyboru elementu listy, element listy wyświetlić do okna logów Log.d.

Do obsługi zdarzenia wykorzystaj kontener umożliwiający nawigację z NavController. Element wybrany wyświetl w oddzielnym oknie, zadбай o możliwość nawigacji między oknami.

Rozbuduj interfejs GUI z informacjami pobranymi z klasy Movie. Skorzystaj z biblioteki io.coil-kt:coil-compose:2.4.0 w celu wizualizacji obrazów pobranych z internetu.

2. Implementacja

Rozszerzenie aplikacji obejmowało dodanie szczegółowego ekranu dla wybranego filmu oraz wprowadzenie komponentów umożliwiających wizualizację obrazów.

Kod Źródłowy został załączony do projektu.

Poniżej opisano kluczowe elementy:

Nowa klasa Movie:

- Dane filmu zawierają pola takie jak:

- title,
- year,
- runtime,
- genere,
- director,
- actors,
- plot
- listę URL-ów do obrazów (images).

- Klasa pozwala na bogatsze przedstawienie informacji o filmie.

DetailsScreen:

- Ekran szczegółowy (DetailsScreen) wyświetla dane wybranego filmu:
- Informacje tekstowe (title, director, plot).
- Obrazy w układzie poziomym (HorizontalScrollableImageView) za pomocą komponentu LazyRow.
- Wykorzystano NavController do nawigacji między ekranami.

Coil Integration:

- Do ładowania obrazów z internetu zastosowano bibliotekę Coil (io.coil-kt:coil-compose:2.4.0).
- Obrazy są ładowane w MovieRow oraz DetailsScreen przy użyciu
- rememberAsyncImagePainter.

Widżety GUI:

MovieRow: Element listy, który pokazuje skrócone informacje o filmie i umożliwia rozwinięcie dodatkowych szczegółów w widoku rozwijanym (*AnimatedVisibility*).

HorizontalScrollableImageView: Wyświetla obrazy filmu w postaci przewijanej listy (*LazyRow*).

TopAppBar: Nagłówek z przyciskiem powrotu.

Udoskonalona nawigacja

NavHost zarządza przejściami między ekranami, w tym przekazaniem *movieId* do *DetailsScreen*.

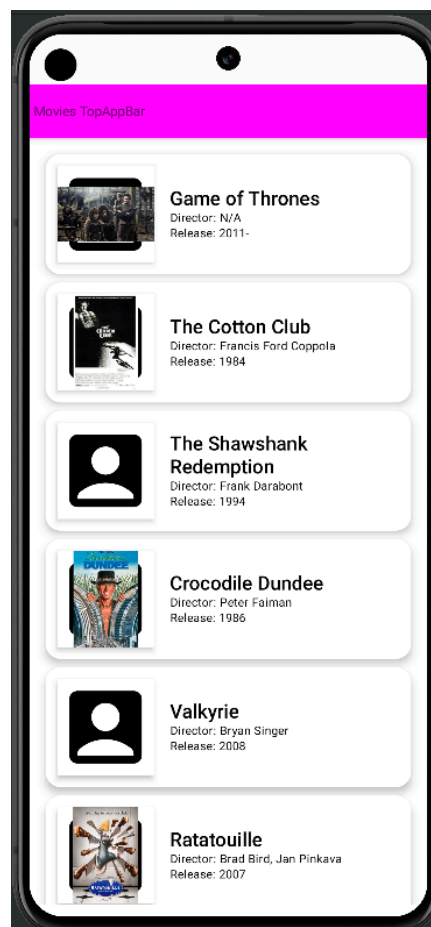
3. Wyniki

Aplikacja działa zgodnie z wymaganiami. Uzyskano następujące funkcjonalności:

- Główny ekran aplikacji wyświetla listę filmów z miniaturą i podstawowymi informacjami.
- Kliknięcie na element listy przenosi użytkownika do szczegółowego ekranu wybranego filmu.
- Szczegółowy ekran wyświetla pełne informacje o filmie, w tym obrazy w układzie przewijanym poziomo.
- Powrót do ekranu głównego jest możliwy za pomocą przycisku w *TopAppBar*.

Poniżej zostaną dodane zrzuty ekranu, ilustrujące:

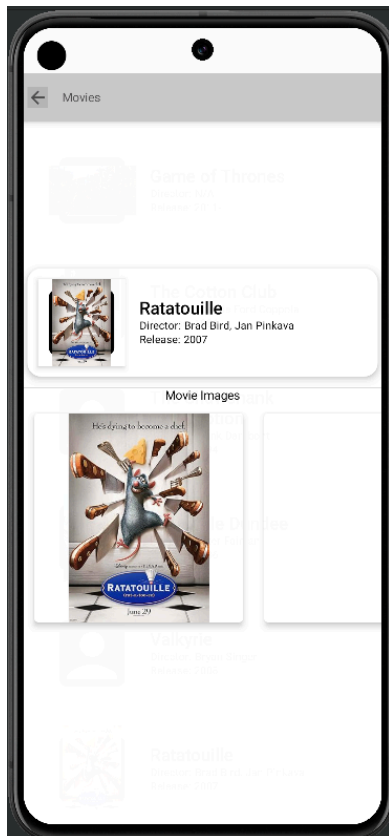
- Ekran główny z listą filmów.



- Szczegółowy ekran filmu z wyświetlanymi informacjami i obrazami.



- Działającą nawigację między ekranami.



4. Podsumowanie

Wprowadzenie klasy Movie oraz integracja z biblioteką Coil znacząco wzbogaciły funkcjonalność aplikacji. Użytkownik może teraz nie tylko przeglądać listę filmów, ale również wyświetlać szczegółowe dane o każdym filmie, w tym obrazy.

Laboratorium pozwoliło na praktyczne zastosowanie zaawansowanych technik Jetpack Compose, takich jak:

- Dynamiczna nawigacja między ekranami.
- Animacje (AnimatedVisibility).
- Integracja z bibliotekami zewnętrznymi do ładowania obrazów.

Aplikacja spełnia założone cele i stanowi dobrą bazę do dalszego rozwoju.