

PROGRAMOWANIE OBIEKTOWE - PROJEKT

OGÓLNY OPIS PROJEKTU

Celem projektu jest stworzenie aplikacji symulującej działanie systemu VOD. Podstawowymi bytami w tym świecie powinny być filmy, seriale, live streaming, użytkownicy i dystrybutorzy.

Każdy produkt w systemie powinien posiadać swoje zdjęcie, nazwę, opis, datę produkcji, czas trwania, dystrybutora, kraj(e) produkcji i ocenę użytkowników. Seriale i filmy mogą być sensacyjne, dramatyczne, komediowe, dla dzieci, dokumentalne lub akcji. Filmy posiadają listę aktorów, linki do zwiastunów, cenę oraz czas w którym można wielokrotnie obejrzeć film po zakupie. Seriale mają cenę, listę aktorów, sezony i odcinki. Każdy odcinek ma datę premiery i długość wyrażoną w minutach. Live streaming to wydarzenie jednorazowe z datą wyświetlenia i jednorazową ceną. Filmy i live streaming mogą być objęte promocją. Promocja charakteryzuje się datą rozpoczęcia, datą zakończenia i procentowym upustem. Upust procentowy musi być w zakresie od 5% do 50%.

Produkty dostępne w systemie są wykupywane przez użytkowników (klientów). Każdy użytkownik jest scharakteryzowany przez imię/nazwę/kod, datę urodzenia, email i numer karty kredytowej. Produkty można wykupywać jednorazowo lub na zasadzie miesięcznie opłacanego abonamentu. Abonament jest dostępny w trzech wersjach: Basic, Family i Premium. Wersje abonamentu różnią się od siebie ceną, liczbą urządzeń, na których można równocześnie odtwarzać treści oraz maksymalną dostępną rozdzielczością produktów. Wydarzenia live streaming mogą być wykupywane jedynie jednorazowo (abonament ich nie obejmuje). Filmy i seriale są objęte abonamentem, ale może je też wykupić użytkownik bez abonamentu. Użytkownik bez abonamentu może w ramach jednej transakcji wykupić jeden film lub jeden odcinek serialu.

Treści do systemu VOD dostarczane są przez dystrybutorów. Każdy dystrybutor zawiera z właścicielem systemu VOD umowę licencyjną. Na podstawie umowy właściciel systemu jest zobowiązany przelewać na konto bankowe dystrybutora środki za obejrzone przez użytkowników treści. Kwota płacona dystrybutorowi jest ustalana jako stała kwota za każde odtworzenie lub w formie ryczałtu miesięcznego.

WYMAGANIA

DOSTĘPNE FUNKCJE I PODSTAWOWE ZAŁOŻENIA:

- użytkownik wprowadza do systemu seriale, filmy, live streaming, użytkowników (klientów) i dystrybutorów poprzez **panel kontrolny**;
- dystrybutorzy są tworzeni tylko na polecenie użytkownika programu, kliencie i produkty mogą pojawiać się automatycznie (proporcjonalnie do liczby produktów w systemie);

- tworzone seriale/filmy/live streaming/klienci/dystrybutorzy powinny mieć automatycznie wylosowane wszystkie wartości pól (nawet jeśli są tworzone na polecenie użytkownika);
- w oknie aplikacji powinna się znajdować lista wszystkich produktów w systemie;
- lista produktów powinna mieć opcję **wyszukiwania tekstowego** według nazwy lub aktorów;
- wybranie produktu z listy powinno skutkować wyświetleniem informacji o wybranym elemencie wraz z **wykresem oglądalności w czasie**;
- każdy klient oraz każdy dystrybutor to **osobny wątek**;
- wątek klienta odpowiada za generowanie w losowych odstępach czasu zamówień i/lub wyświetleń;
- wątek dystrybutora w losowych momentach wypuszcza nowe produkty i renegocjuje umowę;
- użytkownik programu może ręcznie ustalać ceny abonamentu i produktów;
- gdy przez trzy miesiące z rzędu system będzie **generował straty, symulacja się kończy**;
- użytkownik może oglądać podstawowe informacje o obiekcie (produkcje, kliencie, dystrybutorze) w osobnym **oknie lub panelu informacyjnym** po wybraniu obiektu z listy;
- korzystając z przycisków w oknie informacyjnym obiektu, użytkownik może **usunąć** obiekt z systemu;
- do rozwiązania problemów z wielowątkowością należy wykorzystać **semafor lub monitor**;
- powinna istnieć możliwość zapisania i odtworzenia stanu symulacji poprzez **serializację**;
- podstawowymi kryteriami oceny są jakość kodu i funkcjonalność zgodna z wymaganiami; walory estetyczne wizualizacji działają na plus oddającego, ale brak wymyślnej grafiki w żaden sposób nie obniża oceny;
- podobnie jak walory estetyczne, choć niewymagane, na plus będą działać nawiązania do kultury popularnej (np. Netflix, IPLA, Amazon Prime, Telemaniak, Legenda telewizji).

DODATKOWE WYMAGANIA:

- w terminie najpóźniej do **25. listopada 2018 r.** należy **przesłać** prowadzącemu swoją wizję modelowanego świata w postaci **diagramu klas UML**;
- w terminie najpóźniej do **5. grudnia 2018 r.** należy (na zajęciach laboratoryjnych) przedstawić prowadzącemu szkielety **klas Java** napisane zgodnie z zaproponowanym diagramem klas UML;
- wszystkie niestatyczne pola klas powinny być prywatne, a dostęp do nich powinien być realizowany poprzez **setter**y i **getter**y;
- kod projektu powinien być udokumentowany tak by umożliwić wygenerowanie dokumentacji w formacie HTML przy pomocy narzędzia **javadoc**;
- nadesłany projekt powinien zawierać aplikację w postaci **uruchamialnego** pliku *.jar, folder ze źródłami oraz prosty plik *readme* zawierający:
 - imię, nazwisko, numer indeksu, grupę i dzień zajęć,
 - krótką instrukcję obsługi programu;
- kod źródłowy programu wraz z dodatkowymi plikami powinien zostać przesłany na adres prowadzącego **do 27.01.2019 do 23:59**; po tym terminie maksymalna ocena z projektu będzie co tydzień obniżana o ocenę w dół lub wymagane będzie rozbudowanie projektu;

- istnieje możliwość pertraktacji niektórych punktów wymagań funkcjonalnych, ale wszelkie zmiany w projekcie muszą być konsultowane z prowadzącym;
- zastrzega się możliwość wezwania studenta w celu osobistego wyjaśnienia wątpliwości dotyczących programu zaliczeniowego;
- **wszelkie plagiaty będą skutkowały oceną 2,0 zarówno dla osoby udostępniającej jak i kopiującej kod; wykrycie plagiatu po sprawdzeniu projektu i wystawieniu z niego oceny powoduje zmianę oceny na 2,0.**

