Plan zarządzania konfiguracją – System Zarządzania Wypożyczalnią Filmów

# 1. Zakres dokumentu

Dokument definiuje sposób identyfikacji, wersjonowania, kontroli, audytu oraz rejestrowania konfiguracji w projekcie MovieRental. Obejmuje komponenty aplikacji WPF oraz serwera ASP.NET WebAPI.

# 2. Referencje

- Plan projektu  
- Analiza wymagań  
- SAD – projekt aplikacji  
- Plan QA  
- Opis wersji aplikacji (SVD)  
- Repozytorium Git projektu

# 3. Terminologia dokumentu

- CI – Configuration Item  
- CSCI – Computer Software Configuration Item  
- CCB – Configuration Control Board  
- Baseline – zatwierdzony zestaw wersji  
- Build – zbudowana wersja aplikacji

# 4. System zarządzania konfiguracją

- Repozytorium GitHub (lub lokalne Git)  
- Branch: main, develop, feature/...  
- CI/CD ręczne (kompilacja, migracja bazy, uruchomienie testów)

# 5. Proces zarządzania konfiguracją

# 5.1 Identyfikacja konfiguracji (CI)

# 5.1.1 Struktura produktu.

- WPF klient (Lista3.View, Lista3.ViewModel, Lista3.Model)  
- Serwer API (RentalWebApi, MovieRental.Data)  
- Baza danych EF Core  
- Dokumentacja projektowa (docx)

# 5.1.2 Wybór elementów konfiguracji (CSCI)

- RentalListViewModel, RentalRepository, RentalsController, ReportsWindowViewModel

# 5.1.3 Dokumentowanie elementów konfiguracji

- Każdy CI jest zarejestrowany w strukturze katalogów Git  
- Każda zmiana dokumentowana w commit message i changelogu

# 5.1.4 Wersje

- Wersja 1.0.0 – pełna funkcjonalność (CRUD + raporty)  
- Wersjonowanie semantyczne: MAJOR.MINOR.PATCH

# 5.1.5 Definicja zbioru baseline

- Każde release z tagiem Git stanowi baseline  
- Przykład: v1.0.0 (release 23.06.2025)

# 5.2 Kontrola konfiguracji

- Zmiany zatwierdzane przez wykonawcę projektu  
- Commity muszą przechodzić review  
- Merge do main tylko po przetestowaniu

# 5.3 Rejestracja stanu konfiguracji

- Git log i Git tags  
- Historia wersji aplikacji opisana w changelogu  
- Raporty SVD i QA uzupełniane po każdej wersji

# 5.4 Audyt konfiguracji

# 5.4.1 Audyt funkcjonalny

- Porównanie specyfikacji (SAD) z działaniem aplikacji  
- Testy pokrywające przypadki użycia

# 5.4.2 Audyt fizyczny

- Sprawdzenie zgodności katalogów i wersji plików z repozytorium  
- Weryfikacja spójności projektu i dokumentacji

# 6. Organizacja procesu zarządzania konfiguracją

- Właściciel konfiguracji: autor projektu  
- Odpowiedzialny za oznaczanie wersji, utrzymanie repozytorium i wersji dokumentacji

# 7. Procedury zarządzania konfiguracją

# 7.1 Procedura identyfikacji konfiguracji

- Nazwy plików, wersje, identyfikatory Git  
- Foldery: ClientApp/, WebAPI/, Docs/

# 7.2 CCB

- Nieformalna – wszystkie decyzje podejmowane przez autora projektu

# 7.3 Procedura kontroli konfiguracji

- Zmiany w kodzie zatwierdzane przez pull request lub commit  
- Dokumentacja aktualizowana przed wypuszczeniem wersji

# 7.4 Procedura rejestracji stanu konfiguracji

- Rejestracja przez Git + plik changelog.txt  
- Wersje kodu + wersje dokumentów trzymane razem

# 7.5 Procedura audytu konfiguracji

- Checklisty porównawcze: CI z SAD vs obecne repo  
- Sprawdzenie poprawności działania i zgodności z wymaganiami

# 7.6 Harmonogram zarządzania konfiguracją

- Tydzień 2–3: utworzenie struktur CI  
- Tydzień 4–7: kontrola zmian, dokumentacja  
- Tydzień 8: audyt końcowy i baseline

# 8. Kontrola procesu zarządzania konfiguracją

- Regularne przeglądy commitów  
- Porównanie z wymaganiami  
- Raporty QA i changelog jako dowód zgodności