**Specificatiile programului**

1. **Creare Joc Video**:

Utilizatorii trebuie să poată crea un obiect JocVideo specificând numele, platforma și anul lansării. Aceste date trebuie să fie inițializate la crearea obiectului.

1. **Adăugare Faze**:

Utilizatorii trebuie să poată adăuga faze la un joc video. O fază nu poate fi null (ar genera o excepție de tipul ArgumentNullException).

Fazele sunt stocate într-o listă ordonată în care sunt adăugate.

1. **Adăugare Scoruri**:

Utilizatorii pot adăuga scoruri pentru un joc, iar scorurile nu pot fi negative (ar genera o excepție de tipul ArgumentException).

Scorurile sunt stocate într-o listă, iar accesul la această listă este doar pentru adăugare, nu și pentru modificare directă.

1. **Afișare Faze**:

Jocul trebuie să ofere posibilitatea de a afișa toate fazele existente. Dacă nu există faze adăugate, trebuie afișat doar numele jocului.

1. **Verificare Compatibilitate Platformă**:

Se poate verifica dacă jocul este disponibil pe o anumită platformă. Această verificare trebuie să fie insensibilă la cazul literelor (majuscule/minuscule).

Dacă platforma specificată este null, trebuie generată o excepție de tipul ArgumentNullException.

Se așteaptă să se afișeze un mesaj adecvat dacă jocul este sau nu este disponibil pe platforma căutată.

1. **Reprezentare Textuală a Jocului**:

Trebuie să existe o metodă ToString care returnează un șir de caractere formatat cu numele jocului, platforma și anul lansării. Metoda trebuie să funcționeze corect chiar și pentru valori extreme ale acestor proprietăți (de exemplu, valori goale sau negative).

**Specificatiile pentru EQ Partitioning si Boundary Values**

**Adăugare Faze**:

**Equivalence Partitioning**:

Input valid: O șir de caractere non-null și non-gol (de exemplu, "Tutorial"). Acest input ar trebui să fie adăugat cu succes la lista de faze.

Input invalid: Un null sau șir gol ar trebui să declanșeze o excepție ArgumentNullException.

**Boundary Value Analysis**:

Testați cu un șir de caractere gol ("") și un șir de o singură literă ("A") pentru a verifica comportamentul la limita minimă a lungimii unui nume de fază validă.

**Adăugare Scoruri**:

**Equivalence Partitioning**:

Scoruri valide: Orice număr întreg pozitiv sau zero.

Scoruri invalide: Numere întregi negative, care ar trebui să declanșeze o excepție ArgumentException.

**Boundary Value Analysis**:

Testați cu 0 (valoare minimă validă), -1 (puțin sub limita minimă), și int.MaxValue (valoare extremă validă superioară).

**Verificare Compatibilitate Platformă**:

**Equivalence Partitioning**:

Platformă compatibilă: Testați cu input identic cu platforma jocului, dar cu diverse combinații de majuscule și minuscule (de exemplu, "PC", "pc", "Pc").

Platformă incompatibilă: Platforme care nu sunt listate ca suportate (de exemplu, "Xbox", "PlayStation").

**Boundary Value Analysis**:

Nu se aplică în mod direct pentru acest caz deoarece input-ul este textual și nu numeric. Totuși, se poate considera testarea cu valori limite cum ar fi string gol "" sau null.

**Afisare Faze**:

**Equivalence Partitioning**:

Listă cu faze: Când există faze adăugate.

Listă goală: Când nu există faze adăugate, ar trebui afișat doar titlul.

**Boundary Value Analysis**:

Testați cu lista exact după adăugarea primei faze (de exemplu, cu o singură fază), și cu lista complet goală.

**Boundary Value Analysis**

**AdaugaScor**

* **Valorile la limită valide**:
  + Scorul minim valid (zero).
  + Exemplu: 0
  + Scoruri foarte apropiate de un threshold specific, threshold minim: 0 iar threshold maxim: maxim\_value\_integer.
* **Valorile la limită invalide**:
  + Scorul imediat sub minimul valid (imediat sub zero).
  + Exemplu: -1
* **Valorile valide:**
  + Scorurile pozitive din intervalul (0, maxim\_value\_integer).

**Funcția VerificaCompatibilitate(string platformaCautata)**

**Clase de Intrare pentru Partitionarea Echivalenței:**

**Platformele acceptate: “PC”, “Xbox”, “PlayStation”, “Nintendo Switch”, “WII U”.**

* **Validă**:
  + Platforma căutată este identică sau diferă doar prin majuscule/minuscule față de platforma jocului.
  + Exemple: Dacă platforma jocului este "PC", atunci clasele de intrare ar putea include "PC", "pc", "Pc".
* **Invalidă**:
  + Platforma căutată nu se regăsește în lista de platforme acceptate de joc și este diferită de platforma jocului.
  + Exemple: Dacă platforma jocului este "PC", atunci "Xbox", "PlayStation" sunt intrări invalide.
  + Platforma căutată este **null**.

**Funcția AdaugaScor(int scor)**

**Clase de Intrare pentru Partitionarea Echivalenței:**

* **Validă**:
  + Scoruri non-negative, inclusiv zero.
  + Exemple: 0, 1, 50, 100, etc.
* **Invalidă**:
  + Scoruri negative.
  + Exemple: -1, -10, -100, etc.

**Clase de Intrare pentru Analiza Valorilor la Limită:**

* **Valorile la limită pentru scoruri valide**:
  + Scorul zero este o valoare la limită care ar trebui să fie testată separat pentru a confirma comportamentul la această frontieră.
  + Un scor foarte mare, **int.MaxValue**, pentru a testa gestionarea valorilor mari.
* **Valorile la limită pentru scoruri invalide**:
  + Un scor negativ imediat sub limita de zero, cum ar fi -1, este important de testat pentru a verifica aruncarea excepției adecvate.