

## 5. Tétel

*Egy grafikus, több platformon (desktop, web, mobil) is használható kártyajáték programot kell készíteni.*

- *Tegyen javaslatot a feladathoz megfelelő programozási nyelv kiválasztására!*
- *Hasonlítsa össze a választott programozási nyelvet más nyelvekkel, és indokolja választásának előnyeit!*
- *Határozza meg a többplatformú alkalmazásfejlesztés kritikus pontjait és korlátait!*
- *Válasszon ki egy fejlesztési környezetet és indokolja döntését!*

### **Tegyen javaslatot a programozási nyelv és a fejlesztési környezet kiválasztására!**

Játékfejlesztésre számos nyelv közül válogathatunk, mindegyik különböző, számos előnnyel, illetve hátránnyal, hogy említsek párat: C++, Java, C#, Python, JavaScript, Swift.

Hogy melyiket választjuk az már teljesen rajtunk múlik, a tudásunkon. Ha még kezdők vagyunk a játékfejlesztésben, akkor a Python az elsők közt merülhet fel, mivel akik nem is értenek annyira a programozáshoz is áttudják azt látni, hátránya, hogy funkcióiban limitált a játékok fejlesztésére. A java egy sokoldalúbb nyelv és több haladóbb funkciót támogat anélkül, hogy sokkal nehezebbé tenné a kódolást.

### **Hasonlítsa össze a választott programozási nyelvet más nyelvekkel, és indokolja választásának előnyeit!**

A java mellett még egy nagyon népszerű nyelv játékfejlesztésre a C++, mindkét nyelv platform független. A Javat applikációk fejlesztésére használják. Széles körben használják windows alapú, vállalati vagy akár telefonos alkalmazásokhoz. Ezzel szemben a C++-t főképp rendszerszoftverek fejlesztésére.

A C++ a Java-val szemben támogatja a Goto kifejezést, amit érdemes elkerülni, mivel, ha folyamatosan használjuk, akkor gyorsan összezavaróvá válhat a kódunk.

A java-nak számos előnnyel rendelkezik, mint például, hogy egy viszonylag könnyen tanulható nyelv, rendelkezik az ún. WORA-val (*Write once and run everywhere*), vagyis, hogy bármilyen eszközön lefuttathatjuk a programunkat op. rendszertől függetlenül. Illetve támogatja az objektum orientált programozást, amivel átláthatóbb lesz a programunk, illetve legközelebbi projektekhez is majd feltudjuk használni.

### **Határozza meg a többplatformú alkalmazásfejlesztés kritikus pontjait és korlátait!**

Az ún. Seller Lock-in, ez az egyik legjelentősebb kritikus pontja a többplatformú játékfejlesztésnek. Ez azt foglalja magába, hogyha egyik op. rendszerből a másikba szeretnéd átvinni a kódodat, akkor nem biztos, hogy működni fog, nagy eséllyel újra kell majd írd.

### **Válasszon ki egy fejlesztési környezetet és indokolja döntését!**

Fejlesztői környezetekből is hasonlóan választhatunk, mint a programozási nyelvénél, rengeteg van belőlük és a tapasztalatunkon múlik, hogy melyiket érdemes választanunk. Egyik legnépszerűbbek közé tartozik a Unity.

Mivel az egyik legnépszerűbb fejlesztői környezet, emiatt hatalmas közösséggel rendelkezik. Folyamatosan kapja az újabb frissítéseket, funkciókat. Egy viszonylag könnyen tanulható környezettel rendelkezik, ebben segítenek a már előre létrehozott sablonok. Széleskörben rendelkezik támogatott eszközökkel, a webes játékoktól kezdve, a telefonoktól egészen a nagyköltségvetésű játékoktól a konzolokig bezárólag.

Ezek mellett rendelkezik hátrányokkal azokból a lényegesebb, hogy a létrehozott játékok jelentéktelenül sok memóriát vesz igénybe, ami egy gyengébb számítógépen nem annyira elenyésző probléma.