

1. Megvalósítási Dokumentáció

1. Bevezetés

A H5 megvalósítási dokumentáció célja, hogy teljes körű és részletes áttekintést adjon az Infinite Loopers játék tényleges implementációjáról, a rendszer működéséről, az egyes komponensek kapcsolatáról, valamint a fejlesztési folyamat során alkalmazott módszerekről és technológiákról. Ez a fázis jelenti az egész féléves projekt munka gyakorlati megvalósulását, amely során az előző fázisokban meghatározott követelmények, specifikációk és tervelemek a gyakorlatban is implementálásra kerültek.

A fejlesztés egy iteratív, sprint-alapú, agilis folyamatban zajlott. Minden iterációban egy-egy funkciócsoportot valósítottunk meg: kezdve az alap játékmenettel, majd az akadályok működésével, pontozási rendszerrel, kezelőfelülettel, végül pedig a projekt egyedi oktatási funkciójával, a matematikai feladatokat tartalmazó rendszerrrel. Minden fejlesztési ciklus teszteléssel és hibajavítással zárult, amely biztosította a stabil prototípus létrejöttét és a hibák időben történő feltárását.

A játék megvalósított verziója egy stabil, jól strukturált és könnyen karbantartható prototípus, amely nemcsak szórakoztató endless runner élményt nyújt, hanem a matematikai ismeretek játékos fejlesztésére is képes.

2. Az architektúra megvalósítása

Az Infinite Loopers architektúrája három, egymástól világosan elkülönülő rétegre épül. Ez a felépítés lehetővé teszi a modulok közötti lazább csatolást, a könnyebb hibakeresést, valamint az egyszerű bővítést a jövőben.

A rétegek:

1. Felhasználói felület (UI réteg)
2. Játékmenet és logika (logikai réteg)
3. Adatkezelési réteg

Mindhárom réteg külön-külön is fejleszthető és tesztelhető, a GameManager modul pedig biztosítja, hogy ezek harmonikusan működjenek együtt.

2.1 Felhasználói felület (UI réteg)

A felhasználói felület Pygame grafikus moduljaira építve készült el. A megjelenés teljes mértékben igazodik a H4 UI-tervben meghatározott grafikai elvárásokhoz. A UI több különálló képernyőből áll, amelyeket a GameManager kezel és jelenít meg.

A legfontosabb UI komponensek:

- Főmenü – A játék indulásakor megjelenő menü, amelyben a felhasználó kiválaszthatja a játék indítását, a beállításokat vagy a kilépést. A gombok vizuális visszajelzéssel reagálnak a kiválasztásra.
- Beállítások menü – Lehetővé teszi a játékos számára a hangerőszint módosítását, a zene és effektek hangerejét, valamint az egyéb választható paramétereket.
- HUD (Heads-Up Display) – Játék közben folyamatos visszajelzést nyújt az aktuális pontszámról, high score-ról, sebességről, a matematikai feladat szövegéről és a játék előrehaladásáról.
- Game Over képernyő – Ütközés esetén jelenik meg, kiírja a játékos eredményét, a rekordot, és lehetőséget ad újraindításra vagy visszatérésre a főmenübe.

A felhasználói felület folyamatos frissítése garantálja, hogy a HUD és a menüelemek valós időben reagálnak a játékon belüli eseményekre.

2.2 A játékmenet logikai rétege

A logikai réteg az Infinite Loopers motorja, amely meghatározza a karakter mozgását, az akadályok működését és a játék dinamikáját. Ebben a rétegben találhatók meg azok az algoritmusok és vezérlőmodulok, amelyek a játék működését irányítják.

Főbb elemek:

- Player modul – A játékos karakter mozgásáért, ugrásáért, gravitációjáért és animációjáért felel. A karakter különböző állapotok között vált (fut, ugrik, esik, ütközik), amelyet egy állapotgép szabályoz.
- ObstacleManager – Feladata az akadályok generálása, mozgatása és eltüntetése. Időzített algoritmus alapján jeleníti meg az akadályokat, amelyek gyorsulnak, ahogy nő a játék nehézsége.

- MathChallengeManager – Ez az oktatási funkció modulja. A játék időközönként egy matematikai műveletet jelenít meg (pl. „ $8 \times 4 = ?$ ”), és három akadályt generál, amelyek közül csak az egyik tartalmazza a helyes választ. A játékosnak a megfelelő akadálynak kell nekiugrania.

- GameManager – A legfontosabb központi modul. Kezeli a bemeneteket, a játékhurok működését, az állapotváltásokat (menü → játék → game over), valamint összehangolja a többi modult.

- ScoreSystem – A pontszám növeléséért, jutalmazásért és High Score mentésért felel.

Ez a réteg alkotja a játék mechanikai és játékmeneti alapját.

2.3 Adatkezelési réteg

A játék egyszerű fájlalapú adatkezelést alkalmaz, mivel nincs szükség komplex adatbázisra. A rendszer gyors, könnyen módosítható és áttekinthető.

Tárolt adatok:

- highscore.txt – Tartalmazza a játék során elért legmagasabb pontszámot.
- config.json – Opcionálisan tartalmazza a beállításokat (hangerő, felbontás).

A ScoreSystem kezeli az adatok betöltését és mentését. Ha a játékos új rekordot ér el, azonnal frissül a fájl tartalma.

3. Komponensek és osztályok részletes bemutatása

A játék implementációja több modulra lett szétosztva. Minden modul egyértelműen meghatározott felelősségi körrel rendelkezik.

Player osztály:

- Tulajdonságok: pozíció, sebesség, gravitáció, animációs állapot, hitbox.
- Metódusok: update(), jump(), apply_gravity(), detect_collision().
- Animációk és sprite-ok kezelésére sprite sheet-eket használ.

Obstacle és ObstacleManager:

- Véletlenszerű akadálygenerálás algoritmikus úton.
- Nehézségfüggő gyorsulás és ritkítás.
- Ütközések kezelése, akadályok eltávolítása.

MathChallengeManager:

- Matematikai feladat generálása.
- Pályán megjelenő válasz-objektumok létrehozása.
- Hibás választás → Game Over, helyes válasz → jutalmazás.

ScoreSystem:

- Pontszám növelése távolság + idő + helyes válasz alapján.
- High Score kezelése, fájlba írása.

UI modul:

- Menülogika.
- HUD frissítése.
- Felhasználói visszajelzések vizuális megjelenítése.

4. Tervezési minták alkalmazása

A projekt során több klasszikus szoftverfejlesztési mintát alkalmaztunk:

Singleton:

A ScoreSystem egyetlen globális példányban létezik, így minden modul ezt használja a pontkezeléshez.

State Machine:

A Player és a GameManager működését állapotgép vezérli. Ez biztosítja, hogy a karakter mozgása és a játékmenet logikai váltásai tisztán és hibamentesen működjenek.

MVC alapú szerkezet:

- Model: Player, Obstacle, MathChallenge, ScoreSystem
- View: HUD, menük, vizuális elemek
- Controller: GameManager, Input Controller

5. Felhasznált technológiák és eszközök

Programozási nyelvek: Python 3.9+

Könyvtárak:

- Pygame: grafika, eseménykezelés, hangok
- Math: matematikai műveletekhez
- JSON: beállítás"