**ShadowStride – Fejlesztési Dokumentáció**

**I. Projektismertető és Bevezetés**

A **ShadowStride** egy látványos, kihívásokkal teli 2.5D-s akció-platformer videójáték, amelyet a fejlesztőcsapat a **Unity** játékmotor és **C#** programozási nyelv használatával hozott létre. A projekt egy sötét, misztikus, árnyékokkal teli világba kalauzolja a játékost, aki egy árnyékszerű harcost irányít. A játék célja egy olyan élmény megteremtése volt, amely modern vizuális stílust ötvöz a klasszikus platformer játékmenet élvezetével, inspirációt merítve olyan címekből, mint a *Hollow Knight* és az *Ori and the Blind Forest*.

**A Projekt Fő Komponensei**

A ShadowStride fejlesztése rendszerszintű integrációt igényelt, mivel a projekt három fő, különböző technológiákon alapuló részből áll:

1. **Weboldal (React):** A játék leírását, bemutatását és a launcher letöltési lehetőségét kínáló reszponzív felület.
2. **Launcher (Electron):** Asztali alkalmazás a játék indításához, verzióellenőrzéshez és automatikus frissítéskezeléshez.
3. **Játék (Unity, C#):** A játszható 2.5D platformer, a tényleges játékmenet magja.

**II. A Projekt Célja, Motivációja és Játékmenet**

**2.1 Célkitűzések**

A ShadowStride fő célja egy komplex, többplatformos rendszer demonstrálása volt, amely sikeresen integrálja a webes és a játékmotoros technológiákat. A projekt egyben portfólióprojektként és vizsgamunkaként is szolgált, amely bemutatja a csapat együttműködési képességét, az agilis fejlesztési módszerek alkalmazását és a különböző fejlesztői környezetek kezelését. A technikai kihívások mellett kiemelt cél volt egy **könnyen kezelhető, mégis kihívásokkal teli** játékélmény létrehozása.

**2.2 A Játékmenet Összefoglalása**

A játék egy árnyékszerű harcos kalandjait mutatja be, aki szörnyekkel és akadályokkal küzd meg, miközben több, tematikusan eltérő pályán halad keresztül. A játékmenet kulcsa az **ügyesség, a gyors reakció és a stratégiai mozgás**.

**Karakterképességek:**

* **Mozgás:** Futás, ugrás, falmászás (falra tapadás).
* **Harcrendszer:** Karddal való támadás és védekezés.

**Pályadizájn:** A játék több szintre épül (pl. ködös erdő, omladozó romváros, vertikális toronyszerkezet), mindegyik új kihívásokat, csapdákat és ellenfeleket tartogatva. A pályák tartalmaznak **checkpointokat** és rejtett területeket is.

**Ellenfelek Típusai:**

* **Közelharci szörnyek:** Lassú, alapvető támadásokkal.
* **Repülő entitások:** Távolsági támadók.
* **Főellenségek:** Komplex támadási mintákkal és több életsávval.

**III. Technikai Háttér és Rendszerarchitektúra**

A projekt technológiai választásai a modernitást és a hatékonyságot tükrözik, lehetővé téve a komponensek közötti zökkenőmentes kommunikációt.

**3.1 Weboldal – React**

A weboldal a játék elsődleges megjelenési felülete.

* **Cél:** Játékbemutató és a launcher letöltésének biztosítása.
* **Technológiák:** **React 18** (komponensalapú felépítés), **Tailwind CSS** (reszponzív, modern design), **React Router** (oldalnavigáció).
* **Telepítés:** Netlify vagy GitHub Pages alapú statikus publikáció.

**3.2 Launcher**

A launcher a játék indításának és karbantartásának központi eleme.

* **Cél:** Egyszerű indítás, automatikus frissítés, verzióellenőrzés.
* **Architektúra:** C# WPF alkalmazás.
* **Működés:** A Node.js modulok kezelik a **fájlrendszert** és a szerveroldali **API hívásokat** (frissítések), míg az Electron az UI (React) és a natív asztali funkciók (pl. Unity futtatható állomány indítása) közötti kapcsolatot biztosítja.

**3.3 Játékfejlesztés – Unity + C#**

A játék motorja a Unity, amely a 2.5D mechanika alapját képezi.

* **Motor:** **Unity 2022.3 LTS**.
* **Programozás:** **C#** (játéklogika, karaktervezérlés, AI).
* **2.5D Megoldás:** A játék 3D környezetben készült, ami lehetővé teszi a látványos kamerakezelést és mélységérzetet, miközben a karakterek mozgása és az interakciók szigorúan 2 dimenzióban zajlanak, megőrizve a klasszikus platformer irányítást.

**IV. Játékdesign és Látványvilág**

A ShadowStride esztétikája a sötétségre, a kontrasztra és a hangulatos atmoszférára épül.

**4.1 Grafikai Stílus és Atmoszféra**

A vizuális dizájn **letisztult, mégis hangulatos**. Az árnyékszerű harcos és a sötét környezet dominál.

* **Vizuális elemek:** Kézzel rajzolt háttértextúrák, dinamikus **fény-árnyék effektek** és a **Particle Systems** használatával készült részecskeeffektek, amelyek a feszültséget és a felfedezés örömét hivatottak átadni.
* **Karakter és Animáció:** A karakter mozgása és animációi gondosan lettek megtervezve, hogy a játék érzete **természetes és dinamikus** legyen (pl. falra tapadás, támadás).

**4.2 Tartalomterv**

A játék tartalma több, tematikusan megkülönböztetett pályára oszlik:

1. **Ködös Erdő (Kezdő szint):** Főleg horizontális mozgás.
2. **Omladozó Romváros:** Komplexebb platform elemek.
3. **Toronyszerű Struktúra:** Vertikális mozgást és falmászást hangsúlyozó pálya.
4. **Barlang / Árnyéksíkság:** További, tervezett szintek, amelyek bevezetése a jövőbeli fejlesztések célja.

**4.3 Zenei és Hangeffektek**

Az **atmoszférikus háttérzene** és az egyedi **hangeffektek** (támadás, ugrás, szörnyek) hozzájárulnak a játék feszült, misztikus hangulatához. A hanganyagok saját készítésű és szabadon felhasználható források kombinációjából állnak.

**V. Fejlesztési Folyamat, Tesztelés és Ütemterv**

A fejlesztés során az **agilis szemléletet** és a szoros csapatmunkát helyeztük előtérbe.

**5.1 Fejlesztési Eszközök és Metodika**

* **Projektmenedzsment:** **Trello** a feladatok követésére és priorizálására.
* **Verziókezelés:** **Git és GitHub** biztosította a kódbázis szinkronizálását és a fejlesztői ágak kezelését. **Unity Collab** segítette a Unity projektek közös szinkronizálását.
* **Kommunikáció:** Discord a gyors és hatékony megbeszélésekhez.

A projekt **heti sprintekben** haladt, ahol minden iterációban egy-egy fő funkcionális egység (pl. mozgásrendszer, harcrendszer, React weboldal alapja) készült el és került tesztelésre.

**5.2 Tesztelés**

A projekt stabilitásának és minőségének biztosítása érdekében több szintű tesztelést végeztünk:

* **Egységtesztek:** Főleg a C# komponensek (pl. AI, harcrendszer logika) funkcionális ellenőrzése.
* **Manuális Játéktesztelés:** A játékmenet Unity Editorban és buildelt verzióban történő átfogó próbája.
* **Kompatibilitási Tesztek:** A React weboldal különböző böngészőkön, a launcher pedig több operációs rendszeren (Windows, macOS) került ellenőrzésre.

**5.3 Műszaki Specifikációk**

| Kategória | Részletek |
| --- | --- |
| **Fejlesztői Környezet** | Unity 2022.3 LTS, Visual Studio 2022, Node.js 18+, React 18 |
| **Programozási Nyelvek** | C# (Játék), JavaScript (Weboldal, Launcher) |
| **Verziókezelés** | Git, GitHub |
| **Grafikai Eszközök** | Photoshop, Aseprite |

**VI. Jövőbeli Fejlesztések és Összegzés**

**6.1 További Fejlesztési Lehetőségek**

A ShadowStride jelenlegi verziója egy stabil prototípus, amely kiváló alapot ad a további bővítésekhez.

**Tervezett Jövőbeli Funkciók:**

* **Mentési Rendszer:** A játékállás rögzítése és betöltése.
* **Karakterfejlődés:** Karakterfejlődési fa bevezetése, új fegyverek és speciális képességek támogatása.
* **Közösségi Elemek:** Online **ranglista** és **achievement** rendszer integrálása.
* **Lokalizáció:** A játék fordítása több nyelvre.