# **Dokumentácia k hre Arkanoid**

# **Tvorba Levelov hry (Grid)**

Pre tvorbu levelov máme vytvorené prefaby blokov, ktoré sa vkladajú do scény. Bloky v scéne sú zarovnané do mriežky pomocou komponentu Grid.

Ako prvé si v scéne vytvoríme prázdny objekt a pridáme komponent Grid. S požadovanými hodnotami pre veľkosť bunky v mriežke a medzerami medzi bunkami.

Obrázok, na ktorom je text, snímka obrazovky, písmo, číslo

Automaticky generovaný popis

Po kliknutí na objekt s komponentom Grid sa nám v pravom dolnom rohu scény zobrazí tlačidlo pre otvorenie tile palety.

Obrázok, na ktorom je snímka obrazovky, text, multimediálny softvér, softvér

Automaticky generovaný popis

Po zakliknutí tlačidla sa nám zobrazí okno, kde si zvolíme GameObject Brush ako typ štetca.

Obrázok, na ktorom je text, snímka obrazovky, softvér, multimediálny softvér

Automaticky generovaný popis

Pridáme si element do poľa buniek a do kolónky Game Object vložíme prefab bloku ktorý chceme vložiť do scény. Pomocou nástrojov na vrchu okna môžeme „kresliť“ bloky do scény, vymazávať ich a pod.

# **Skóre a hviezdičky**

V hre existujú dva typy skóre. Klasické číselné skóre a hviezdičky.

Klasické skóre sa v hre pripočítava spôsobom, že pri zničený bloku, blok vo funkcii OnDisable() vyvolá event, ktorý informuje skript LevelManager, že blok bol zničení a ako argument eventu sa posiela skóre, ktoré daný blok pridáva.

Obrázok, na ktorom je text, písmo, snímka obrazovky, grafika

Automaticky generovaný popis

LevelManager následne skóre pripočíta k celkovému skóre a skontroluje či v scéne existujú ešte ďalšie bloky. Ak bol zničený posledný blok, skóre sa odošle skriptu GameManager a načíta sa scéna WinScreen.

Obrázok, na ktorom je text, snímka obrazovky, softvér, písmo

Automaticky generovaný popis

Hviezdičky sa počítajú jednoduchým spôsobom, za každý stratený život v levely sa hráčovi odoberie 1 hviezdička. Túto hodnotu reprezentuje premenná currentLives, ktorá je taktiež posielaná skriptu GameManager.

# **Level Editor**

V scéne level editora sa nachádza objekt Grid podľa ktorého sa v scnéne ukladajú bloky položené hráčom a objekt Background, ktorého sprite sa mení na základe pozadia zvoleného hráčom.

Obrázok, na ktorom je text, písmo, snímka obrazovky, dizajn

Automaticky generovaný popis

V skripte LevelEditor sa vo funkcii Update() získava pozícia myši na obrazovke, ktorá s konvertuje na súradnice bunky objektu komponentu Grid a kontroluje kliknutie pravého a ľavého tlačidla myši.

Obrázok, na ktorom je text, snímka obrazovky, softvér

Automaticky generovaný popis

Po kliknutí ľavým tlačidlom myši sa zavolá funkcie PlaceBlock() a po kliknutí pravým tlačidlom myši sa zavolá funkcia DeleteBlock()

Vo funkcii PlaceBlock() sa skontroluje či pokladáme blok na miesto, kde už blok existuje, ak áno, tak sa blok zničí a na jeho miesto a položí nový, ak nie tak sa nový blok iba na dané súradnice položí.

Obrázok, na ktorom je text, snímka obrazovky, softvér, písmo

Automaticky generovaný popis

Vo funkcii DeleteBlock() sa vykoná to isté ako v prvej časti funkcie PlaceBlock(), čo vlastne znamená že to je duplicitný kód a táto funkcia mohla byť využitá vo funkcii PlaceBlock(), ale tak čo už.

Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis

Bloky sú vyberané pomocou funkcie SelectBlock(), ktorá je volaná pomocou tlačidiel v UI hry.

Obrázok, na ktorom je text, písmo, snímka obrazovky, grafika

Automaticky generovaný popis

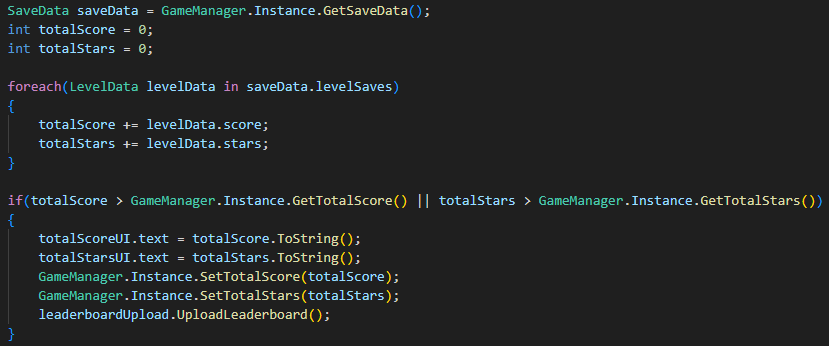
Pozadie sa mení pomocou funkcie ChangeBackground(), ktorá ako argument príma integer smeru, -1 alebo 1, keďže vyberanie pozadia v UI je reprezentované pomocou šípok tieto hodnoty reprezentujú šípku, ktorú používateľ stlačil. Následne sa pomocou tejto hodnoty posúvame v poli pozadí a nastavujeme vybrané pozadie ako sprite objektu Background.

Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis

# **Leaderboard**

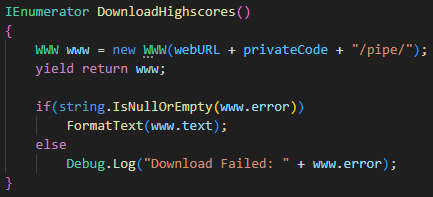
Skript TotalScoreCalculator, ktorý sa nachádza na objekte v scéne LevelSelection sa stará o spočítanie kompletného skóre a všetkých hviezdičiek, ktoré v hre hráč získal. Tieto dáta sú načítané zo súboru SaveData. Ak je počet hviezdičiek alebo skóre väčšie ako predchádzajúce maximálne hodnoty tak sa tieto hodnoty updatnú v UI aj v Leaderboard. Leaderboard sa updatne zavolaním funkcie UpdateLeaderboard() v skripte LeaderboardUpload.



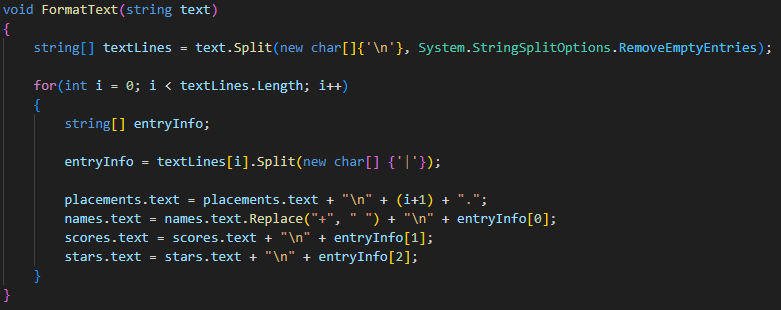
Pre uchovávanie dát leaderboard sa využíva online služba dreamlo.com. Pomocou API služby sa odosiela meno hráča, skóre a počet hviezdičiek. Na volanie API sa používa trieda WWW z knižnice UnityEngine, ktorá je však teraz už deprikovaná.



O získanie dát z leaderboard sa stará skript LeaderboardDownload, kde sa znovu pomocou API v coroutine funkcii získajú dáta zo služby vo forme textu.



Tento text sa následne vo funkcii FormatText rozdelí podľa riadkov a údaje sa zobrazia v UI scény Leaderboard.



# **Pohyb loptičky**

Pohyb loptičky je riešený nastavovaním premennej velocity na komponente RigidBody. Hodnota velocity je vypočítaná vynásobením normalizovaného vektora (smeru pohybu) a rýchlosti, ktorá je konštantná.

Smer pohybu loptičky sa kalkuluje pri odraze od padla alebo od steny vo funkcii OnCollisionEnter2D().

Ak je kolíznym objektom padlo vypočíta sa smer odrazu od padla podľa minimálnych a maximálnych povolených uhľov odrazu a vzdialenosti bodu kolízie od stredu padla. Tento spôsob odrážania loptičky od padla simuluje odrážanie v originálnej hre Arkanoid, kde sa smer odrazu určuje podľa toho na ktorú stranu padla loptička dopadla a uhol odrazu sa určuje podľa vzdialenosti bodu kolízie od stredu.

Obrázok, na ktorom je text, snímka obrazovky, písmo, softvér

Automaticky generovaný popis

Ak je kolíznym objektom čokoľvek iné, smer odrazu sa kalkuluje pomocou funkcie Reflect(). Ak by však uhoľ odrazu bol väčší ako 160 stupňov (odraz by bol príliš blízko kolmému odrazu), uhol odrazu sa nastaví na hodnotu 160 stupňov.

Následne sa nastaví premenná velocity.

Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis

# **Pohyb padla**

Pohyb padla je kalkulovaný na základe smeru, ktorým sa myš pohybuje v osi X (nie na základe pozície myši). Takýmto spôsobom sa vyhneme problémom kedy sa padlo zastaví na kraji obrazovky avšak myš pokračuje ďalej v pohybe, čo spôsobuje, že padlo musí „čakať“, kedy sa myš vráti na hrateľnú obrazovku. Taktiež nám to umožňuje kontrolovať rýchlosť akou sa padlo pohybuje.

Pozícia padla sa nastavuje vypočítaním hodnoty amountToMove vynásobením smeru pohybu a relatívnou rýchlosťou, ktorou chceme aby sa padlo pohybovalo. Táto hodnota sa následne pripočíta k hodnote x pozície padla.

Obrázok, na ktorom je snímka obrazovky, text, písmo

Automaticky generovaný popis