STŘEDNÍ ŠKOLA TECHNICKÁ A EKONOMICKÁ BRNO,

OLOMOUCKÁ, PŘÍSPĚVKOVÁ ORGANIZACE



MATURITNÍ PRÁCE

Lost World - 2D platform game

Brno 2025 Marek Šedivý

# Čestné prohlášení

Prohlašuji, že předložená maturitní práce pod názvem Lost World - 2D platform game je ve smyslu autorského zákona výhradně mým autorským dílem. Maturitní práci jsem vypracoval samostatně s použitím uvedené literatury a dalších uvedených zdrojů na základě konzultací s vedoucím maturitní práce a oponentem.

Datum Podpis

# Poděkování

Děkuji vedoucímu mé práce panu Mgr. Zdeňku Brokešovi za odborné vedení, cenné rady a podporu během vývoje této hry. Dále děkuji všem, kteří se podíleli na testování a poskytli zpětnou vazbu, což pomohlo hru vylepšit.

# Zadání

Potvrzené zadání ředitelem školy (zástupcem ředitele, vedoucím sekce IT, garantem).

# Anotace

Práce se zabývá návrhem a vývojem 2D plošinové hry Lost World v herním enginu Unity v programovacím jazyce C#. Hra je založena na překonávání překážek, sbírání mincí a eliminaci nepřátel pomocí střelby ohnivých koulí. Hráč se pohybuje v šesti úrovních, kde využívá skoky, jetpack, slidování po zdech a další mechaniky.

Obsah

[Čestné prohlášení 2](#_Toc189087266)

[Poděkování 3](#_Toc189087267)

[Zadání 4](#_Toc189087268)

[Anotace 5](#_Toc189087269)

[Úvod – vymezení tématu a cílů práce 9](#_Toc189087270)

[Teoretická část 10](#_Toc189087271)

[Vývojové prostředí Unity 10](#_Toc189087272)

[Historie a vývoj Unity 10](#_Toc189087273)

[Proč byl zvolen pro vývoj hry Unity 10](#_Toc189087274)

[Základní struktura Unity pro 2D hry 10](#_Toc189087275)

[Platformové hry a jejich principy 11](#_Toc189087276)

[Co je platformová hra 11](#_Toc189087277)

[Typické herní mechaniky v platformových hrách 11](#_Toc189087278)

[Význam fyziky v platformových hrách 12](#_Toc189087279)

[Programování v jazyce C# pro Unity 13](#_Toc189087280)

[Význam jazyka C# v herním vývoji 13](#_Toc189087281)

[Komponentový systém Unity 13](#_Toc189087282)

[Správa objektů, animací a fyziky pomocí C# 13](#_Toc189087283)

[Herní fyzika a kolize v Unity 13](#_Toc189087284)

[Fyzikální engine v Unity 13](#_Toc189087285)

[Rigidbody 2D a jeho vlastnosti 14](#_Toc189087286)

[Kolize a jejich detekce v Unity 14](#_Toc189087287)

[Ladění fyziky a kolizí 14](#_Toc189087288)

[Grafika a animace v 2D hrách 15](#_Toc189087289)

[Význam grafiky v herním vývoji 15](#_Toc189087290)

[Tvorba herní grafiky v aplikaci Piskel 15](#_Toc189087291)

[Import spritů do Unity 15](#_Toc189087292)

[Animace v Unity 16](#_Toc189087293)

[Vizuální efekty ve hře 16](#_Toc189087294)

[Shrnutí a optimalizace grafiky 16](#_Toc189087295)

[Praktická část 18](#_Toc189087296)

[Návrh hry 18](#_Toc189087297)

[Herní koncept a základní mechaniky 18](#_Toc189087298)

[Herní úrovně a jejich design 18](#_Toc189087299)

[Nepřátelé a překážky 18](#_Toc189087300)

[Sbírání mincí a systém achievementů 19](#_Toc189087301)

[Herní menu a uživatelské rozhraní 19](#_Toc189087302)

[Shrnutí návrhu hry 19](#_Toc189087303)

[Implementace hlavních mechanik 19](#_Toc189087304)

[Pohyb hráče 19](#_Toc189087305)

[Detekce kolizí a ověřování, zda je hráč na zemi 20](#_Toc189087306)

[Použití jetpacku 21](#_Toc189087307)

[Střelba ohnivých koulí 21](#_Toc189087308)

[Kolize s nepřáteli a poškození hráče 22](#_Toc189087309)

[Statické platformy s možností propadu 22](#_Toc189087310)

[Systém sběru mincí a úspěchů 23](#_Toc189087311)

[Sběr mincí a jejich ukládání 23](#_Toc189087312)

[Systém úspěchů 24](#_Toc189087313)

[Shrnutí implementace systému sběru mincí a úspěchů 25](#_Toc189087314)

[Herní UI a ovládání 26](#_Toc189087315)

[Hlavní prvky uživatelského rozhraní 26](#_Toc189087316)

[Implementace základních UI prvků 26](#_Toc189087317)

[Ukazatel paliva pro jetpack 27](#_Toc189087318)

[Počítadlo nasbíraných mincí 28](#_Toc189087319)

[Herní menu 28](#_Toc189087320)

[Pauzovací menu 29](#_Toc189087321)

[Obrazovka Game Over 30](#_Toc189087322)

[Závěr 31](#_Toc189087323)

[Seznam použité literatury 32](#_Toc189087324)

[Seznam použitých značek a symbolů 33](#_Toc189087325)

[Seznam použitého software 34](#_Toc189087326)

[Seznam obrázků 35](#_Toc189087327)

[Seznam příloh 36](#_Toc189087328)

[Přílohy 37](#_Toc189087329)

# Úvod – vymezení tématu a cílů práce

Cílem práce bylo vytvořit 2D plošinovou hru inspirovanou klasickou hrou Super Mario, tedy platform game, kde hráč překonává překážky, sbírá mince a bojuje s nepřáteli. Hra byla vytvořena v herním enginu Unity, přičemž programování probíhalo v jazyce C#.

Hlavní důvod volby Unity bylo jeho široké a praktické využití pro vývoj 2D her a také možnost efektivní práce s herní fyzikou, animacemi a zvukem.

Hra obsahuje šest úrovní, zatímco první z nich slouží jako úvod do hry, který seznamuje hráče s ovládáním a základními mechanikami hry.

Postava se dokáže pohybovat, skákat, útočit ohnivými koulemi po nepřátelích, sklouzávat po zdech a propadávat skrze platformy.

Při vývoji bylo nutné zrealizovat několik oblastí jako fyzika postavy, detekce kolizí a optimalizace výkonu hry.

Herní grafika byla vytvořena zcela vlastními prostředky, a to pomocí aplikace Piskel, která je určena pro tvorbu pixel-artové grafiky. Veškeré vizuální prvky, včetně prostředí, postav, animací a uživatelského rozhraní, byly navrženy tak, aby odpovídaly zvolenému stylu a zároveň zajišťovaly vizuální přehlednost a konzistenci hry.

# Teoretická část

## Vývojové prostředí Unity

Herní engine Unity je jedním z nejpopulárnějších nástrojů pro vývoj videoher. Používá se jak profesionálními vývojáři, tak začátečníky díky své uživatelsky přívětivé struktuře, podpoře více platforem a rozsáhlé komunitě. Unity umožňuje tvorbu 2D i 3D her, přičemž hra Lost World byla vyvíjena v jeho 2D režimu.

### Historie a vývoj Unity

Unity bylo poprvé představeno v roce 2005 jako herní engine určený pro macOS, ale postupně se rozšířilo na různé platformy včetně Windows, Linux, Android, iOS a WebGL. Díky flexibilnímu komponentovému systému a podpoře jazyka C# se stalo jedním z nejpoužívanějších nástrojů pro vývoj nezávislých her i komerčních titulů.

### Proč byl zvolen pro vývoj hry Unity

Při vývoji hry Lost World bylo Unity vybráno především díky své podpoře pro 2D herní vývoj, která zahrnuje specializované nástroje pro práci s fyzikou, animacemi a správou sprite grafiky. Jedním z hlavních důvodů volby tohoto enginu je také jeho komponentový systém, který umožňuje snadno upravovat herní objekty a přidávat jim nové funkce prostřednictvím předpřipravených komponent, jako jsou fyzikální vlastnosti nebo animace.

Další výhodou Unity je možnost skriptování v jazyce C#, který je objektově orientovaný a poskytuje dostatek flexibility pro implementaci složitějších herních mechanik. Tento jazyk umožňuje efektivně spravovat herní logiku a usnadňuje práci s fyzikálním systémem enginu.

Velkou výhodou tohoto enginu je také rozsáhlá komunita a široká podpora, díky čemuž je možné snadno nalézt dokumentaci, návody a řešení problémů. Unity disponuje obsáhlými oficiálními zdroji, které poskytují podrobné informace o jednotlivých funkcích enginu. V případě potíží je možné využít také komunitní fóra a tutoriály, což značně usnadňuje proces učení a vývoje her.

Díky těmto vlastnostem se Unity ukázalo jako ideální volba pro vývoj Lost World, neboť umožnilo snadnou práci s herní fyzikou, animacemi, UI a celkovou správou herních objektů.

### Základní struktura Unity pro 2D hry

Unity využívá GameObjecty, které představují základní stavební prvky všech objektů ve hře. Každý GameObject může být doplněn o různé komponenty, které definují jeho vlastnosti a chování. Mezi nejčastěji používané komponenty při vývoji 2D her patří Sprites, což jsou grafické objekty reprezentující herní prvky, jako jsou postavy, nepřátelé nebo prostředí.

Dalším klíčovým prvkem je Rigidbody 2D, který umožňuje fyzikální simulaci objektů, například gravitační působení na hráče či nepřátele. Pro detekci kolizí mezi herními objekty se využívají Colliders, mezi které patří například BoxCollider2D nebo CircleCollider2D. Tyto komponenty určují, jak se objekty vzájemně dotýkají a jak na sebe reagují při střetu.

Pro tvorbu prostředí se v Unity často používá systém Tilemaps, který umožňuje efektivní práci s dlaždicovými mapami a usnadňuje vytváření rozsáhlejších úrovní. Vizuální stránku hry doplňuje Animator, který se stará o plynulé přechody mezi jednotlivými animacemi postav a objektů. Kromě těchto nástrojů Unity obsahuje i UI System, který umožňuje vytváření uživatelského rozhraní, například ukazatele zdraví, počítadla mincí nebo herního menu.

Díky těmto funkcím Unity poskytuje vývojářům komplexní nástroje pro tvorbu 2D her a umožňuje snadnou správu herních mechanik, fyziky i vizuálních efektů.

## Platformové hry a jejich principy

### Co je platformová hra

Platformová hra, známá také jako platformer, je žánr videoher, ve kterém hráč ovládá postavu pohybující se po různých platformách. Cílem hráče bývá zpravidla překonávání překážek, sbírání předmětů a vyhýbání se nebo porážení nepřátel. Hry tohoto typu jsou často založeny na přesném ovládání pohybu postavy, zejména na skákání mezi platformami a vyhýbání se překážkám.

Většina platformových her používá 2D perspektivu, kde se hráč pohybuje vodorovně (doleva/doprava) nebo svisle (nahoru/dolů). Existují však i 3D platformovky, které nabízejí větší volnost pohybu ve všech směrech. Mezi klíčové vlastnosti platformových her patří jednoduché, ale přesné ovládání, které hráči umožňuje reagovat na dynamické situace v herním prostředí.

### Typické herní mechaniky v platformových hrách

Platformové hry jsou charakteristické specifickými herními mechanikami, které určují způsob, jakým se hráč pohybuje a interaguje s herním prostředím. Pohyb postavy je jedním ze základních prvků, který ovlivňuje hratelnost. Hráč má možnost se pohybovat různou rychlostí, běhat nebo chodit, přičemž zrychlení a setrvačnost často závisí na konkrétním herním designu.

Dalším důležitým prvkem je skákání, které umožňuje překonávání překážek a přechod mezi jednotlivými platformami. V některých platformových hrách je implementován také dvojitý skok, který hráči poskytuje větší kontrolu nad pohybem. Pokročilejší tituly obsahují také mechaniky jako wall-jump, tedy odraz od stěny, který umožňuje dosáhnout míst, kam by se hráč jinak nedostal.

Interakce s prostředím je dalším zásadním prvkem platformových her. Hráč se může setkat s pohyblivými plošinami, propadajícími se podlahami nebo různými pastmi, které testují jeho reflexy a přesnost ovládání. Kromě překážek se v platformových hrách běžně vyskytují i sběratelské předměty, jako jsou mince, klíče nebo power-upy. Tyto předměty mohou sloužit jako prostředek k dosažení vyššího skóre nebo poskytovat hráči různé výhody, například dočasnou nesmrtelnost, vyšší rychlost nebo větší sílu útoku.

V některých platformových hrách je také zakomponován bojový systém, který umožňuje hráči eliminovat nepřátele různými způsoby. Některé hry umožňují útoky nablízko, zatímco jiné využívají střelné zbraně nebo magické schopnosti. V některých případech lze využít i prostředí samotné, například shazováním objektů na nepřátele nebo využíváním speciálních pastí.

Dalším důležitým aspektem platformových her je herní fyzika, která ovlivňuje chování postavy i objektů v herním světě. Gravitační síly, rychlost pádu nebo odpor vzduchu mohou výrazně změnit hratelnost. Správně nastavená fyzika je klíčová pro plynulé a přirozené ovládání, které dává hráči pocit kontroly nad postavou.

V případě hry Lost World byly implementovány všechny základní platformové mechaniky, přičemž některé z nich byly rozšířeny o nové prvky. Hráč se může pohybovat, skákat, používat jetpack, střílet ohnivé koule a interagovat s různými typy plošin. Herní svět je navržen tak, aby postupně učil hráče nové mechaniky a zároveň zajišťoval dostatečnou výzvu.

### Význam fyziky v platformových hrách

Herní fyzika hraje zásadní roli v platformových hrách, protože ovlivňuje realistické chování postav a objektů v herním světě. Unity využívá fyzikální engine Box2D, který umožňuje simulaci gravitace, kolizí a pohybové dynamiky.

Každý herní objekt, který se pohybuje nebo reaguje na fyzikální síly, musí mít přiřazenou komponentu Rigidbody 2D, která určuje jeho hmotnost, rychlost a gravitační působení. Spolu s komponentami Colliders pak Unity umožňuje detekci kolizí mezi objekty, což je důležité pro správnou funkci pohybu postavy, překážek a nepřátel.

V platformových hrách se často nastavují specifické fyzikální hodnoty, například nižší gravitace pro plynulejší skákání nebo větší odpor vzduchu, aby pohyb působil realisticky.

Hra Lost World využívá fyzikální engine Unity k simulaci pohybu postavy, jetpacku a interakce s prostředím. Bylo nutné správně nastavit hmotnost postavy, gravitační sílu a kolizní systémy, aby byla hratelnost plynulá a zábavná.

## Programování v jazyce C# pro Unity

### Význam jazyka C# v herním vývoji

Programovací jazyk C# je jedním z nejdůležitějších prvků při vývoji her v Unity. Jde o objektově orientovaný jazyk, který umožňuje efektivní správu kódu a využívání herních mechanik prostřednictvím skriptů. C# byl vyvinut společností Microsoft a díky své flexibilitě a snadné čitelnosti se stal oblíbeným nástrojem nejen pro vývoj aplikací, ale i her.

V Unity se C# používá především k ovládání herních objektů, detekci kolizí, správě uživatelského rozhraní a fyzikálních interakcí. Každý herní objekt v Unity může obsahovat skript v C#, který definuje jeho chování a interakce s prostředím.

### Komponentový systém Unity

Unity pracuje na principu komponentového systému, což znamená, že každý herní objekt (GameObject) je tvořen základní strukturou, do které mohou být přidány různé komponenty. Tyto komponenty určují chování a vlastnosti daného objektu.

Každý herní objekt může obsahovat několik základních komponent, jako jsou:

* Transform – Uchovává informace o poloze, rotaci a velikosti objektu.
* Sprite Renderer – Zajišťuje zobrazení grafického prvku (postava, mince, nepřítel).
* Rigidbody 2D – Přidává fyzikální vlastnosti, jako je gravitace nebo pohyb.
* Collider 2D – Umožňuje detekci kolizí mezi objekty.
* Script (C#) – Definuje vlastní chování objektu.

V Unity lze vytvářet vlastní komponenty pomocí C# skriptů. Tyto skripty lze následně přiřadit libovolným objektům ve hře.

### Správa objektů, animací a fyziky pomocí C#

V Unity lze skripty využít nejen pro pohyb postavy, ale i pro správu animací, uživatelského rozhraní a fyzikálních interakcí. Při práci s animacemi se často využívá třída Animator, která umožňuje přepínání mezi jednotlivými animacemi na základě vstupů hráče.

Pro detekci kolizí mezi objekty se využívají metody OnTriggerEnter2D() a OnCollisionEnter2D(), které reagují na doteky mezi objekty. Například při dotyku hráče s mincí se mince odstraní ze scény a hráči se přičte skóre.

## Herní fyzika a kolize v Unity

### Fyzikální engine v Unity

Unity obsahuje integrovaný fyzikální engine, který umožňuje simulaci gravitace, pohybu, kolizí a interakce objektů. Při vývoji 2D her Unity využívá Box2D, což je speciální fyzikální systém určený pro dvourozměrné prostředí. Tento engine zajišťuje, že objekty se chovají realisticky na základě nastavených parametrů, jako jsou hmotnost, odpor, tření nebo gravitační síla.

Pro správné fungování fyzikálních simulací je nutné herním objektům přiřadit komponentu Rigidbody 2D, která definuje jejich pohyb a interakci s okolním prostředím. Každý objekt, který má být ovlivněn fyzikou, musí mít tuto komponentu aktivní.

### Rigidbody 2D a jeho vlastnosti

Komponenta Rigidbody 2D se v Unity stará o fyzikální simulaci objektů ve 2D prostředí. Obsahuje několik klíčových parametrů, které ovlivňují chování objektu:

* Mass (hmotnost) – Určuje, jak moc je objekt ovlivněn fyzikálními silami.
* Gravity Scale (měřítko gravitace) – Umožňuje nastavit sílu gravitace působící na objekt.
* Drag (odpor vzduchu) – Snižuje rychlost objektu pohybujícího se vzduchem.
* Freeze Rotation (uzamčení rotace) – Zabraňuje objektu v otáčení kolem své osy.

Při vývoji bylo nutné upravit hodnotu Gravity Scale, aby skákání postavy působilo přirozeně. Pokud by hodnota gravitace byla příliš vysoká, hráč by neměl dostatek času na kontrolu pohybu během skoku. Naopak příliš nízká gravitace by způsobila, že by postava působila jako ve stavu beztíže.

### Kolize a jejich detekce v Unity

Kolize jsou klíčovým prvkem platformových her, protože určují, jak objekty reagují na kontakt mezi sebou. V Unity se k detekci kolizí používají komponenty Collider 2D, které definují tvar a velikost kolizní oblasti objektu. Existuje několik typů kolizních komponent:

* BoxCollider2D – Používá se pro čtvercové a obdélníkové objekty.
* CircleCollider2D – Vhodný pro kruhové objekty, například mince nebo koule.
* PolygonCollider2D – Umožňuje vytvořit složitější tvar kolizní oblasti.

Při vývoji hry bylo nutné správně nastavit kolizní masky, aby se určité objekty mohly srážet a jiné na sebe nereagovaly. Například mince musí být detekovány hráčem, ale nesmí ovlivňovat fyziku ostatních objektů.

### Ladění fyziky a kolizí

Při vývoji hry bylo nutné detailně ladit fyziku a kolizní systémy, aby byla hra plynulá a hráč měl nad postavou dostatečnou kontrolu. Jedním z hlavních problémů bylo správné nastavení kolizí mezi hráčem, plošinami a nepřáteli.

V některých případech se stávalo, že se hráč „zasekl“ na okrajích platformy nebo že skákání nebylo dostatečně plynulé. Tyto problémy byly vyřešeny úpravou kolizních vrstev (Collision Layers) a optimalizací Rigidbody 2D parametrů.

Celkově správné nastavení fyziky a kolizí výrazně ovlivnilo plynulost hry a odezvu ovládání, což je klíčové pro kvalitní platformovou hratelnost.

## Grafika a animace v 2D hrách

### Význam grafiky v herním vývoji

Grafická stránka hry je jedním z nejdůležitějších aspektů, který ovlivňuje celkový vizuální dojem a atmosféru hry. V případě 2D her se obvykle používají sprity, což jsou dvourozměrné bitmapové obrázky reprezentující postavy, nepřátele, prostředí a různé interaktivní objekty. Správné použití grafiky nejenže zlepšuje hratelnost, ale také umožňuje hráči lépe se orientovat v herním prostředí.

V případě hry Lost World byla grafika navržena v pixel-artovém stylu, který se vyznačuje jednoduchostí, ostrými hranami a retro vzhledem. Tento styl byl zvolen kvůli jeho estetice a také kvůli menším nárokům na výkon hry.

### Tvorba herní grafiky v aplikaci Piskel

Při vývoji hry Lost World byla veškerá grafika vytvořena ručně pomocí aplikace Piskel, která je určená pro tvorbu pixel-artových obrázků. Tento nástroj umožňuje vytvářet statické i animované sprity a nabízí uživatelsky přívětivé rozhraní s funkcemi specificky určenými pro práci s pixelovou grafikou.

V aplikaci Piskel byly vytvořeny všechny postavy, nepřátelé, platformy, pozadí a interaktivní objekty, přičemž každý z těchto prvků byl exportován jako samostatný sprite nebo sprite sheet pro animace.

Při navrhování grafiky bylo nutné dbát na konzistentnost velikostí jednotlivých objektů. Ve hře byl použit grid o velikosti 16×16 pixelů, což znamená, že většina objektů byla navržena tak, aby odpovídala tomuto měřítku.

### Import spritů do Unity

Po dokončení návrhu grafiky bylo nutné sprity importovat do Unity. Unity poskytuje vlastní Sprite Renderer, což je komponenta, která umožňuje vykreslování dvourozměrných obrázků ve hře.

Při importu spritů bylo důležité nastavit správné parametry:

* Pixels Per Unit – Nastavení měřítka spritu v jednotkách Unity. Pro správnou velikost se použila hodnota 16 pixels/unit, což zajistilo, že se grafika zobrazí v odpovídajícím měřítku.
* Sprite Mode – Pro animace bylo nutné nastavit režim Multiple, který umožňuje rozdělit sprite sheet na jednotlivé snímky animace.
* Filter Mode – Nastaven na Point (no filter), což zajistilo ostré hrany spritů bez rozmazání.

Každý importovaný sprite byl následně přiřazen odpovídajícímu GameObjectu ve scéně a doplněn o další komponenty, například fyzikální kolize nebo animace.

### Animace v Unity

Animace jsou klíčovým prvkem, který zajišťuje plynulý pohyb postav a objektů. V Unity se k animacím používá Animator, což je vestavěný nástroj umožňující přepínání mezi jednotlivými snímky animace a řízení přechodů mezi nimi.

V případě hry Lost World byly vytvořeny následující animace:

* Animace chůze hráče – Postava mění pozice nohou podle směru pohybu.
* Animace skákání – Při skoku se mění poloha těla postavy a po dopadu se vrací do výchozí polohy.
* Animace střelby – Při použití ohnivé koule se zobrazuje specifická animace útoku.
* Animace nepřátel – Nepřátelé mají cyklický pohyb
* Animace vody – Vizuální efekt tekoucí vody zajišťuje dynamiku prostředí.

V Unity lze animace vytvářet dvěma hlavními způsoby:

* Frame-by-frame animace – Použití sprite sheetů s jednotlivými snímky pohybu.
* Bone-based animace – Použití rigging systému, který umožňuje animaci pomocí kloubového skeletu (vhodné spíše pro 3D postavy).

V případě Lost World byly použity převážně frame-by-frame animace, protože pixel-artový styl vyžaduje přesné přepínání mezi jednotlivými snímky.

### Vizuální efekty ve hře

Kromě animací bylo nutné do hry přidat další vizuální efekty, které zvýrazňují určité herní mechaniky, byly použity při střelbě ohnivé koule, kde se zobrazí malá exploze v místě dopadu, čímž se zvýrazní úspěšný zásah nepřítele.

Unity umožňuje práci s Particle Systems, což je systém částicových efektů, které lze použít například pro kouř, oheň nebo magické útoky. V případě Lost World byl tento systém využit pro vizualizaci ohnivých střel a efektů vody.

### Shrnutí a optimalizace grafiky

Během vývoje bylo nutné dbát nejen na vizuální stránku hry, ale také na její výkonovou optimalizaci. Použití příliš velkých textur by mohlo negativně ovlivnit plynulost hry, proto byly všechny sprity optimalizovány na co nejmenší velikost.

Při práci s animacemi bylo důležité správně nastavit přechody mezi jednotlivými stavy, aby nedocházelo k vizuálním chybám, například přerušovaným pohybům nebo nesprávnému zobrazení snímků.

Hra Lost World díky těmto úpravám dosahuje plynulých animací a konzistentního vizuálního stylu, který odpovídá pixel-art estetice.

# Praktická část

## Návrh hry

### Herní koncept a základní mechaniky

Hra Lost World byla navržena jako 2D plošinovka inspirovaná klasickými tituly, jako je Super Mario Bros. nebo Celeste. Cílem hráče je projít jednotlivé úrovně, vyhnout se překážkám, sbírat mince a porazit nepřátele. Hra obsahuje šest úrovní, přičemž první slouží jako tutoriál, který hráče učí základní mechaniky.

Mezi hlavní herní mechaniky patří pohyb postavy, skákání, používání jetpacku, střelba ohnivých koulí a interakce s prostředím. Hráč může procházet plošinami, vyhýbat se nebezpečným ostnům, bojovat s nepřáteli a sbírat mince, které slouží k odemykání nových skinů postavy.

### Herní úrovně a jejich design

Každá úroveň byla navržena tak, aby postupně zvyšovala obtížnost a hráče učila nové mechaniky. Struktura hry je rozdělena na šest levelů, kde každý přináší nové výzvy.

* Level 1 (Tutoriál) – Hráč se učí základní pohyb, skákání a sbírání mincí.
* Level 2 – Přidává první nepřátele, kteří se pohybují po určité trase.
* Level 3 – Zavádí použití jetpacku pro překonávání širších mezer mezi platformami.
* Level 4 – Obsahuje složitější překážky, například pohyblivé platformy a ostny.
* Level 5 – Kombinuje všechny dosavadní mechaniky a přidává větší výzvy.
* Level 6 (Finální úroveň) – Nejkomplexnější level s různými kombinacemi nepřátel, pastí a plošin.

Při návrhu úrovní bylo důležité dbát na plynulý nárůst obtížnosti, aby hráč měl čas si osvojit jednotlivé mechaniky, než se setká s větší výzvou. Bylo také nutné správně rozmístit checkpointy, aby hráč nemusel po každém neúspěchu začínat úplně od začátku.

### Nepřátelé a překážky

Hra obsahuje dva typy nepřátel. První typem jsou pohybliví nepřátelé, kteří se po dotyku s hráčem způsobí poškození. Druhým typem jsou ostny, které jsou umístěny na zemi a hráč je musí přeskočit.

Pro implementaci nepřátel byla použita komponenta Rigidbody2D, která umožňuje jejich plynulý pohyb po scéně. Kolize s hráčem jsou detekovány pomocí Colliderů a skriptů v jazyce C#, které snižují hráčovo zdraví v případě kontaktu s nepřítelem.

### Sbírání mincí a systém achievementů

Ve hře jsou rozmístěny mince, které hráč může sbírat. Nasbírané mince se ukládají a mohou být využity k odemčení nových skinů pro postavu.

Kromě mincí hra obsahuje systém achievementů, který odměňuje hráče za splnění určitých podmínek:

* Zabíjení nepřátel – Hráč získá achievement po určitém počtu poražených nepřátel.
* Počet smrtí – Pokud hráč zemře určitý početkrát, odemkne speciální achievement.
* Posbírané mince – Pokud hráč nasbírá všechny mince v úrovni, získá odměnu.

Tento systém podporuje znovuhratelnost a motivuje hráče k prozkoumávání úrovní.

### Herní menu a uživatelské rozhraní

Hra obsahuje jednoduché uživatelské rozhraní UI, které zobrazuje klíčové informace pro hráče. V levém horním rohu se nachází ukazatel zdraví, pod ním je počítadlo mincí a napravo je ukazatel paliva.

Herní menu zahrnuje:

* Hlavní menu – Možnost zahájit hru, změnit nastavení nebo ji ukončit.
* Pauzovací menu – Možnost zastavit hru a vrátit se do hlavního menu.
* Nastavení – Umožňuje změnit hlasitost hudby a zvukových efektů.

Grafické prvky menu byly vytvořeny v pixel-artovém stylu pomocí aplikace Piskel.

### Shrnutí návrhu hry

Během návrhu hry bylo nutné vytvořit herní mechaniky, nepřátele, překážky, sběratelské předměty a uživatelské rozhraní. Celý návrh byl zaměřen na vytvoření zábavného a plynulého herního zážitku, který hráče postupně seznamuje s novými výzvami a mechanikami.

## Implementace hlavních mechanik

### Pohyb hráče

Pohyb hráče ve hře Lost World je založen na chůzi, skákání, používání jetpacku, propadání platformami a střelbě ohnivých koulí. Všechny tyto mechaniky jsou řízeny ve skriptu Player\_movement.cs a využívají Rigidbody2D pro fyzikální interakce.

Úryvek kódu z Player\_movement.cs:

Obsah obrázku text, snímek obrazovky, software

Popis byl vytvořen automaticky

Pohyb hráče je řízen proměnnou moveSpeed, skákání proměnnou jumpForce a propadání platformami je umožněno tím, že při stisknutí S se dočasně deaktivuje BoxCollider2D.

### Detekce kolizí a ověřování, zda je hráč na zemi

Skákání funguje pouze tehdy, pokud je hráč na zemi. Toto je kontrolováno pomocí Raycastu, který detekuje kontakt s vrstvou "Ground".

Úryvek kódu z Player\_movement.cs:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Tento kód sleduje, zda se hráč dotýká objektů s tagem "Ground", a podle toho nastavuje proměnnou isGrounded. Pokud hráč není na zemi, není možné provést další skok.

### Použití jetpacku

Jetpack je implementován jako možnost dočasného letu. Hráč může jetpack aktivovat stisknutím klávesy Space, ale spotřebovává se palivo, které se po čase automaticky doplňuje.

Úryvek kódu z Player\_movement.cs:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Jetpack je aktivován, pokud má hráč dostatek paliva. Jakmile je palivo vyčerpáno, hráč nemůže pokračovat v letu a musí počkat na jeho automatické dobíjení.

### Střelba ohnivých koulí

Střelba ohnivých koulí je realizována ve skriptu Fireball\_Shooter.cs. Po stisknutí klávesy Q se vytvoří nový objekt ohnivé koule, který se pohybuje ve směru, kterým je hráč otočen.

Úryvek kódu z Fireball\_Shooter.cs:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Ohnivá koule se generuje v místě firePoint, což je bod umístěný před hráčem.

### Kolize s nepřáteli a poškození hráče

Pokud hráč narazí na nepřítele nebo ostny, jeho zdraví se sníží.

Úryvek kódu z HP.cs:

Obsah obrázku text, snímek obrazovky

Popis byl vytvořen automaticky

### Statické platformy s možností propadu

Platformy v Lost World jsou statické, ale hráč může propadnout skrz ně, pokud stiskne klávesu S.

Úryvek kódu z Platform.cs:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Obrázek

Pokud hráč stiskne S, kolizní box platformy se dočasně vypne, což umožní propadnutí skrz. Jakmile hráč klávesu pustí, kolize se opět zapne.

## Systém sběru mincí a úspěchů

### Sběr mincí a jejich ukládání

Ve hře hrají mince důležitou roli, protože umožňují hráči odemknout nové skiny postavy. Systém sbírání mincí byl implementován pomocí Collider2D a OnTriggerEnter2D, kde se při dotyku hráče s mincí objekt odstraní a hodnota nasbíraných mincí se zvýší.

Úryvek kódu z Coins\_manager.cs:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Tento skript zajišťuje, že se hodnota mincí zobrazí na obrazovce a aktualizuje se v reálném čase.

Sbírání mincí je řízeno metodou OnTriggerEnter2D(), která detekuje kontakt hráče s mincí a následně ji odstraní:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Každá mince má přiřazený tag "Player", který zajistí, že se mince sebere pouze tehdy, když se jí dotkne hráč.

### Systém úspěchů

Hra obsahuje tři hlavní úspěchy, které jsou udělovány hráči na základě jeho výkonu ve hře. Systém úspěchů je implementován ve skriptu Achievements.cs.

Úryvek kódu z Achievements.cs:

Obsah obrázku text, snímek obrazovky

Popis byl vytvořen automaticky

Tento skript uchovává statistiky hráče a aktualizuje je v uživatelském rozhraní (UI). Úspěchy jsou udělovány na základě následujících podmínek:

* Zabití nepřátel – Pokud hráč porazí určitý počet nepřátel, je mu udělen úspěch.
* Počet úmrtí – Pokud hráč umře mnohokrát, zobrazí se speciální odměna.
* Nasbírané mince – Pokud hráč dosáhne určitého počtu mincí, získá speciální ocenění.

V metodě Update() se hodnoty neustále obnovují a zobrazují na obrazovce.

Každý úspěch může být uložen pomocí PlayerPrefs, což umožňuje uchování statistik i po restartu hry:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

### Shrnutí implementace systému sběru mincí a úspěchů

Herní mechanika sběru mincí byla vytvořena pomocí Collider2D a OnTriggerEnter2D, přičemž mince se při dotyku hráče odstraní a přičte se ke skóre. Systém úspěchů sleduje hráčovy úspěchy a poskytuje vizuální zpětnou vazbu o dosažených výsledcích.

Tento systém nejen podporuje znovuhratelnost, ale také motivuje hráče k lepšímu výkonu a průzkumu úrovní.

## Herní UI a ovládání

Herní uživatelské rozhraní (UI) hraje klíčovou roli v poskytování informací hráči během hry. V Lost World se UI skládá z několika prvků, které zobrazují aktuální stav hry, například zdraví hráče, množství paliva pro jetpack a počet nasbíraných mincí. Kromě toho obsahuje hra také hlavní menu, pauzovací menu a obrazovku pro výběr úrovně.

### Hlavní prvky uživatelského rozhraní

* Ve hře se nachází následující hlavní prvky UI:
* Ukazatel zdraví – Počet zbývajících životů hráče
* Ukazatel paliva – Zobrazuje aktuální stav paliva jetpacku
* Počítadlo mincí – Ukazuje počet nasbíraných mincí
* Hlavní menu – Obsahuje možnosti spustit hru, vybrat úroveň nebo ukončit hru
* Pauzovací menu – Možnost zastavit hru a vrátit se do hlavního menu
* Obrazovka Game Over – Zobrazí se, pokud hráč ztratí všechna životy

### Implementace základních UI prvků

V Unity jsou UI prvky vytvořeny pomocí Canvas systému, který umožňuje zobrazování textu, tlačítek a dalších interaktivních prvků na obrazovce.

Ukazatel zdraví se nachází v levém horním rohu a zobrazuje aktuální počet životů hráče. Tato mechanika je implementována ve skriptu HP.cs.

Úryvek kódu z HP.cs:

Obsah obrázku text, snímek obrazovky

Popis byl vytvořen automaticky

Hodnota zdraví se aktualizuje pokaždé, když hráč utrpí poškození. Jakmile hodnota klesne na nulu, spustí se Game Over obrazovka.

### Ukazatel paliva pro jetpack

Ukazatel paliva zobrazuje aktuální množství paliva, které hráč může využít při letu. Tento prvek UI je propojen se skriptem Player\_movement.cs.

Úryvek kódu pro zobrazení paliva:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Hodnota paliva se aktualizuje v reálném čase a zobrazí se na obrazovce. Pokud hráč palivo vyčerpá, jetpack se dočasně deaktivuje.

### Počítadlo nasbíraných mincí

Počet nasbíraných mincí se zobrazuje v pravém horním rohu a je aktualizován při každém sebrání mince.

Úryvek kódu z Coins\_manager.cs:

Obsah obrázku text, snímek obrazovky, Písmo, software

Popis byl vytvořen automaticky

Po každém sebrání mince se hodnota coinCount zvýší a automaticky se aktualizuje UI.

### Herní menu

Hra obsahuje hlavní menu, které se zobrazí při spuštění hry a umožňuje hráči spustit hru, vybrat úroveň nebo ukončit hru.

Úryvek kódu z LevelMenuManager.cs:

Obsah obrázku text, snímek obrazovky, Písmo, software

Popis byl vytvořen automaticky

Po kliknutí na tlačítko Start Game se načte první úroveň. Pokud hráč klikne na Quit Game, hra se ukončí.

### Pauzovací menu

Hráč může během hry pozastavit hru a zobrazit pauzovací menu.

Úryvek kódu z Pause\_menu.cs:

Obsah obrázku text, snímek obrazovky

Popis byl vytvořen automaticky

Při stisknutí klávesy Escape se hra pozastaví a zobrazí se menu s možností pokračovat nebo se vrátit do hlavního menu.

### Obrazovka Game Over

Pokud hráč ztratí všechna životy, zobrazí se Game Over obrazovka s možností restartovat úroveň nebo ukončit hru.

Úryvek kódu z Game\_over.cs:

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

Po smrti hráč může restartovat úroveň nebo se vrátit do hlavního menu.

# Závěr

Text

# Seznam použité literatury

Text

# Seznam použitých značek a symbolů

2D – dvoudimenzionální nebo dvourozměrný a označuje svět, který je možné popsat dvěma rozměry

UI -

# Seznam použitého software

Unity

Visual studio

Piskel

# Seznam obrázků

Jak

# Seznam příloh

Text

# Přílohy

text