

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická

Platformová hra pro Android v Unity engine

Jakub Dřímál

Vedoucí: Ing. Šebek Jiří
Obor: Softwarové inženýrství a technologie
Květen 2021

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Dřimal** Jméno: **Jakub** Osobní číslo: **475383**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra počítačů**
Studijní program: **Softwarové inženýrství a technologie**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Platformová hra pro Android v Unity engine

Název bakalářské práce anglicky:

Platform game for Android in Unity engine

Pokyny pro vypracování:

Cílem projektu je vytvořit plnohodnotnou hru pro Android v Unity engine. Úkolem hráče je dostat se z jednoho místa na druhé.

Cestou hráč potkává různé pasti, protivníky a hráč má možnost s nimi bojovat nebo naopak bojovat. Cestou hráč sbírá různé předměty,

kteří mu mohou pomoci. Obtížnost se postupně zvětšuje.

Seznámte se s Unity enginem, vytvořte příběh a návrh hry, uživatelské rozhraní, implementujte aplikaci, vyobrazte navigaci v obrazovkách hry a chování hráče / nepřátel a jejich animace za pomoci grafové logiky, proveďte uživatelské testy a napište případný vliv jejich zpětné vazby na hru, nahrajte hru na Google Play.

Pokyny:

1. Seznámení se s Unity, srovnání s dalšími enginy
2. Navrhnutí hry, včetně jejího příběhu a designu
3. Implementování aplikace
4. Provedení uživatelských testů, popsání jejich vlivu na hru
5. Nahrání výsledné aplikace na Google Play

Seznam doporučené literatury:

1) Schell, Jesse. The Art of Game Design: A book of lenses. CRC press, 2008.

[<http://www.sg4adults.eu/files/art-game-design.pdf>]

2) Norton, Terry. Learning C# by developing games with unity 3D. Packt Publishing Ltd, 2013.

[<http://www.computo-visual.org/Videojuegos/Libros/Learning%20C%23%20by%20Developing%20Games%20with%20Unity%203D.pdf>]

3) Rollings, Andrew, and Ernest Adams. Andrew Rollings and Ernest Adams on game design. New Riders, 2003.

[<https://pdfs.semanticscholar.org/4a2e/b20f0b92820834637987fb1da04669c6a845.pdf>]

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Jiří Šebek, kabinet výuky informatiky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **21.02.2021**

Termín odevzdání bakalářské práce: **21.05.2021**

Platnost zadání bakalářské práce: **19.02.2023**

Ing. Jiří Šebek
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Poděkování

Rád bych poděkoval panu Ing. Jiřímu Šebkovi za odborné vedení, ochotu, cenné rady, připomínky a za čas, který mi věnoval při vypracování této bakalářské práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 1. května 2021

Abstrakt

Tato práce se zabývá vytvořením mobilní hry pro platformu Android v herním engine Unity. Součástí práce je porovnání nejznámějších herních engineů a podrobnější analýza zvoleného editoru. Dále práce popisuje návrh hry v herním návrhovém dokumentu, rozvržení uživatelského rozhraní a herního světa, společně se seznamem funkčních a nefunkčních požadavků na výslednou aplikaci. Implementační část pojednává o architektuře aplikace a rozboru technologií a postupů užitých při realizaci jednotlivých částí hry. V závěrečné části se práce zabývá provedením uživatelských testů a jejich vlivu na hru, a následném vydání výsledné aplikace v obchodě Google Play.

Klíčová slova: hra, android, unity, mobilní aplikace

Vedoucí: Ing. Šebek Jiří

Abstract

This thesis describes the creation of a mobile game for the Android platform in the Unity game engine. Part of the work is a comparison of the most popular game engines and a more detailed analysis of the selected editor. Furthermore, the work focuses on the design of the game in the game design document, the layout of the user interface and the game world, along with a list of functional and non-functional requirements for the application. Implementation section discusses the architecture of the application and technologies and procedures used during the implementation of individual parts of the game. The final part deals with user tests and their impact on the game and the subsequent release of the final application in the Google Play store.

Keywords: game, android, unity, mobile application

Title translation: Platform game for Android in Unity engine

Obsah

1 Úvod	1	2.3.3 Limbo	9
2 Analýza	3	2.3.4 The Mooseman	10
2.1 Srovnání herních enginů	3	2.4 Porovnání existujících řešení . . .	10
2.1.1 Unreal	3	3 Návrh hry	11
2.1.2 Unity	4	3.1 Herní návrhový dokument	11
2.1.3 GameMaker	4	3.1.1 Koncept	11
2.1.4 Godot	4	3.1.2 Příběh	12
2.2 Popis Unity engine	4	3.1.3 Postavy	12
2.2.1 Rozložení editoru	5	3.1.4 Level design	12
2.2.2 Scéna, kamera	6	3.1.5 Hratelnost	12
2.2.3 Gameobject, prefab	6	3.1.6 Hudba a zvuky	14
2.2.4 Fyzika	6	3.1.7 Ovládání	14
2.2.5 Asset store	7	3.1.8 Uživatelské rozhraní	15
2.3 Existující podobné hry	8	3.2 První spuštění hry	16
2.3.1 Lucid Dream Adventure 3	8	3.3 Funkční a nefunkční požadavky .	16
2.3.2 Typoman mobile	8	3.4 Uživatelské rozhraní	17
		3.5 Diagramy hry	20
		3.6 Části herního světa	22

4 Implementace	23	6 Vydání hry	37
4.1 Architektura	23	6.1 Založení vývojářského účtu	37
4.2 Technologie	25	6.2 Vyplnění informací o hře	37
4.2.1 Uživatelské rozhraní	25	6.3 Vydání hry	38
4.2.2 Pohyb postav	26	7 Závěr	41
4.2.3 Prostředí	26	Literatura	43
4.2.4 Interakce s předměty	27	A Obrázky ze hry	45
4.3 Herní svět	27	B Odkaz na hru	49
4.3.1 Stromy	28	C Projekt	51
4.3.2 Balvan	28		
4.3.3 Bedna	29		
4.3.4 Páčka	30		
4.3.5 Lučištník	30		
4.3.6 Pavouk	31		
5 Testování	33		
5.1 Provedení uživatelských testů ..	33		
5.2 Vliv zpětné vazby na hru	35		

Obrázky

2.1 Vzhled pracovní plochy editoru v základním rozložení [1]	5	3.9 Navigace obrazovek ve hře	20
2.2 Základní scéna s krychlí a kamerou [1]	6	3.10 Rozhodovací diagram postavy Lučištník	21
2.3 Unity Asset Store [1]	7	3.11 Diagram významnějších tříd ve hře.	21
2.4 Lucid Dream Adventure 3 [2]	8	3.12 Vyobrazení herního světa	22
2.5 Typoman mobile [3]	9	4.1 Component diagram herního engine projektu [7]	24
2.6 Limbo [4]	9	4.2 Component diagram distribuce hry	25
2.7 The Mooseman [5]	10	4.3 Stromy na které lze vylézt, pohled ze hry (vlevo) a ve scéně (vpravo) .	28
3.1 Sekvenční diagram logiky ve hře	13	4.4 Vyobrazení kutálejícího se balvanu (ladění)	29
3.2 Use case diagram hráče	15	4.5 Luk mířící na Panáčka (ladění) .	31
3.3 Dostupnost dotykové obrazovky v landscape režimu [6]	17	4.6 Pochod Pavouka (ladění)	31
3.4 Hlavní menu	18	6.1 Webové rozhraní Google Play Console	38
3.5 Ovládání	18	6.2 Výsledná hra v obchodě Google Play	39
3.6 Nastavení	19	A.1 Pochod lesem	45
3.7 Herní scéna	19	A.2 Padající balvan	46
3.8 Pauza	20		

A.3 Pronásledující pavouk.....	46
A.4 Bedna jako obrana přes šípem .	47



Kapitola 1

Úvod

Počítačové hry nás doprovází náš svět téměř od počátků samotné výpočetní techniky [8]. S postupem času a technologického vývoje se hraní her rozšířilo natolik, že se počítačové hry staly běžnou součástí dnešního digitálního světa. Díky dnešním technologiím a chytrým zařízením je možné hrát hry v podstatě kdekoliv - ať už doma, venku, ve veřejné dopravě či v práci.

V posledních letech popularita hraní mobilních her velmi vzrostla. Za poslední desetiletí vzrostl počet uživatelů mobilních zařízení hrající hry z 44 na 57 procent [9]. Téměř každý má dnes k dispozici chytré zařízení, které hraní počítačových her podporuje. Vzniká tak velmi silné odvětví herního průmyslu, které má obrovskou uživatelskou základnu.

Obecně by se dalo předpokládat, že hráči mobilních her jsou většinou příležitostní hráči, kteří hrají hry pár minut, například při čekání na autobus nebo volné chvíli v práci. Není to tak úplně pravda. Při rozsáhlé studii bylo zjištěno, že se průměrná délka hraní pohybuje okolo 15 - 25 minut denně. Více než polovina hráčů hraje hry každý den, přičemž někteří jsou schopni strávit u jedné hry i několik hodin [10]. Statisticky tvoří mobilní hry dvě třetiny tržeb z celkového herního průmyslu a tento podíl se každým rokem zvyšuje [11].

Tato práce se zabývá vývojem hry právě na mobilní zařízení. Jejím cílem je poskytnout hráčovi možnost se odreagovat, zabavit, nebo se jen na chvíli zaneprázdnit. V této práci je popsáno vytváření platformové hry pro operační systém Android. Vývoj je zachycen samotných počátků návrhu, výběru engine, implementace, až po testování a vydání výsledné hry v obchodě Google Play.

Kapitola 2

Analýza

2.1 Srovnání herních enginů

Herní engine je software určený pro vývoj her, který obsahuje sadu funkcí běžně užívaných v počítačových hrách. Součástí takového editoru je sada nástrojů, které mají za úkol vývojářům vytváření hry urychlit a usnadnit.

K zhotovení této práce bylo vzhledem k obsáhlosti projektu vhodné vybrat si již existující herní engine a nevytvářet vlastní, jednak kvůli časové náročnosti projektu, a také kvůli zamezení takzvaného *Reinventing the wheel* [12], neboli vytváření technik a metod, které byly v minulosti již lépe vytvořeny. Volba herního engine je poměrně zásadní fází projektu, protože se toto rozhodnutí ponese po celou dobu realizace projektu a bude mít vliv na mnoho věcí i do budoucna.

2.1.1 Unreal

Unreal engine je open source software vytvořený společností Epic Games, prvně předvedený v roce 1998. Engine je napsán v C++ a těší se oblibě především ve velkých herních studiích. Mezi jeho přednosti patří především široká škála platforem, na které lze projekt exportovat, a také vysoce kvalitní vizuální stránka výsledného produktu. S tímto kladem ovšem souvisí i fakt, že oproti jiným enginům může mít Unreal větší nároky na hardware.

■ 2.1.2 Unity

Multiplatformní engine od společnosti Unity Technologies, prvně představen v červnu roku 2005. Umožňuje vývoj 2D a 3D her, nabízí přívětivou dostupnost pro začínající vývojáře a export do mnoha platforem. Programovacím jazykem je C#. Unity je obecně považováno za herní engine vhodný pro menší studia či jednotlivce. Po Unrealu je Unity druhým nejvíce používaným herním enginem na světě.

■ 2.1.3 GameMaker

Engine původně vytvořený Mark Overmensem podporující 2D vytváření her, s myšlenkou zpřístupnit vývojářům vytvářet hry bez větší znalosti programovacího jazyka. Gamemaker používá vlastní programovací jazyk Game Maker Language a samotný engine patří k jedním z nejjednodušších programů pro vytvoření vlastní hry.

■ 2.1.4 Godot

Godot je poměrně nový komunitně vyvíjený engine, vydaný v lednu roku 2014. Je zde možné vytvářet 2D a 3D hry a programovat je lze v C++, C#, či interním skriptovacím jazykem enginu, zvaným GDScript. Výhodou enginu je, že je tzv. lightweight (má nízké nároky na prostředky počítače), zdarma, a jeho uživatelská základna roste. V současné době může být nevýhodou menší zaostalost použitých 3D technik oproti konkurenčním enginům.

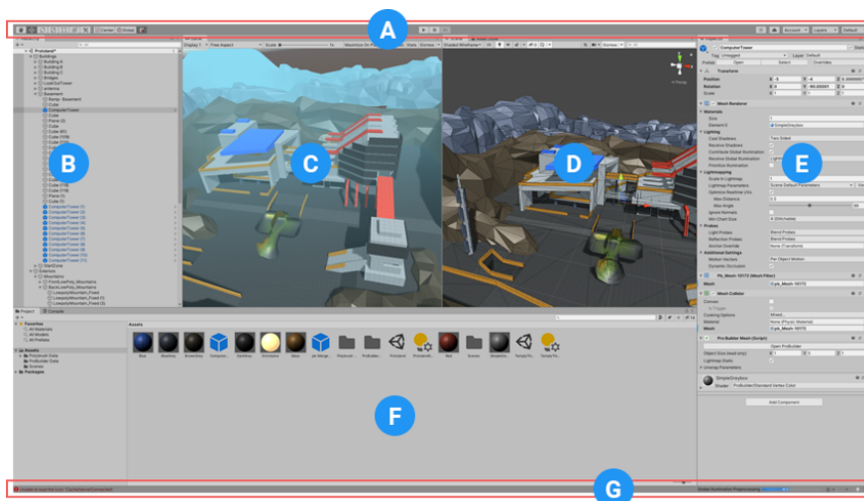
■ 2.2 Popis Unity engine

Jak již název práce napovídá, k vypracování projektu jsem po prozkoumání jednotlivých možností zvolil právě Unity engine. Hlavními důvody tohoto rozhodnutí byly především kompromis mezi nabízenou funkcionalitou a náročností používání enginu, snadné přepínání mezi 2D a 3D zobrazením, nízká náročnost na hardware, programování skriptů v jazyce C#, dostupnost programu a jeho podpora. Unity nabízí čtyři typy licencí software, pod kterými lze engine používat, a to Personal, Plus, Pro, Enterprise [13]. Verze se mezi sebou

liší většinou pouze přídatnou funkcionalitou (jako jsou například analytics, cloud solutions, či technickou podporou) a maximálními příjmy, do kterých lze daná licence používat. Funkce dostupné v samotném enginu ovšem zůstávají stejné ve všech verzích bez ohledu na licenci, což je přívětivé. Pro tuto práci bude zcela dostačovat licence Personal, která je zdarma. Hlavní omezení této licence je splash screen s Unity logem při startu aplikace a výdělek menší než 100 000\$ za rok, což je pro tento projekt zcela přijatelné. Následující kapitola popisuje vlastnosti a možnosti editoru, které Unity nabízí.

2.2.1 Rozložení editoru

Základní rozložení pracovní plochy Unity může vypadat podobně jako na obrázku níže.



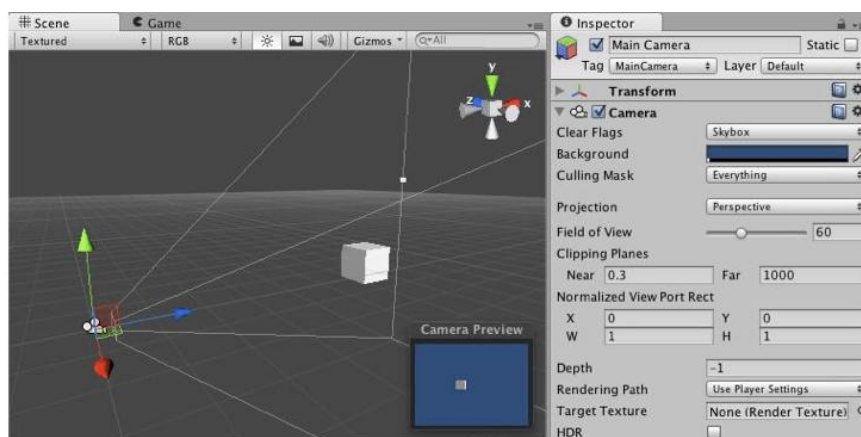
Obrázek 2.1: Vzhled pracovní plochy editoru v základním rozložení [1]

Nahoře je k dispozici panel s nejvíce běžně používanými nástroji (A). Po levé straně se nachází hierarchické okno scény (B), které v sobě obsahuje každý objekt ve aktuální scéně. Největší okna editoru jsou právě ty uprostřed, a tím je pohled na hru (C) a scénu (D). Napravo následuje Inspector (E), který zobrazuje a umožňuje upravit vlastnosti právě vybraného objektu ve scéně. Posledním oknem je Project window (F), zobrazující všechny assety tohoto projektu.

Jednotlivé okna a části lze libovolně schovat či zobrazit, přesouvat, nebo nahrazovat jinými komponentami editoru. V tomto ohledu je pracovní plocha Unity velmi dynamická a vývojář si ji tak může přizpůsobit podle svých potřeb.

2.2.2 Scéna, kamera

Scéna v Unity obsahuje jednotlivé objekty hry a její prostředí. Každou scénu si lze představit jako individuální obrazovku hry, úroveň, či menu. Hra tedy může mít více scén, mezi kterými lze libovolně přepínat. Nepostradatelným objektem scény je kamera. Kamera je zařízení, přes které se hráč dívá na svět. Ve scéně je možné mít i více kamer a mezi nimi přepínat, přičemž každá kamera může mít různé nastavení - první například může herní svět projektovat perspektivně a druhá ortograficky.



Obrázek 2.2: Základní scéna s krychlí a kamerou [1]

2.2.3 GameObject, prefab

GameObject je základním objektem pro veškeré entity ve scéně. Slouží jako kontejner pro jednotlivé komponenty, které na něj lze navěšovat a které mu přidávají další funkcionalitu. Prefab je systém dědění objektů, který nám dovoluje vytvořit z daného GameObjectu znovupoužitelný asset. Jedná se o jakousi šablonu, ze které můžeme vytvářet nové instance a následně je upravovat její vlastnosti. Změnou vlastností původního objektu lze tak měnit všechny objekty, které jej využívají. To nám umožňuje efektivně pracovat s objekty a lépe je organizovat v jednotlivých scénách.

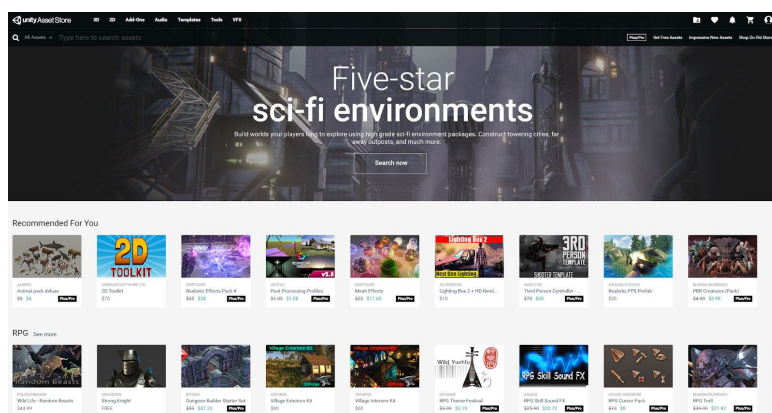
2.2.4 Fyzika

Abychom nemuseli vytvářet celý fyzikální systém od začátku, nabízí nám Unity možnost využít již některé vytvořené fyzikálních komponenty pro naše

objekty. To nám usnadní mnoho práce a času, který můžeme využít na tvorbu jiných částí hry. Vytváření vlastní implementace těchto komponent od začátku by totiž trvalo minimálně několik týdnů, ne-li měsíců [14]. Jednou ze součástí tohoto systému je Collider, což je komponenta zaměřená na kolize objektu. Objektu, na kterém je přidána, vyhradí určitý tvar, který bude sloužit jako kolizní obálka. Je zde k dispozici více typů Colliderů pro různé tvary objektů, jako například BoxCollider, CapsuleCollider, MeshCollider a další. Ne vždy platí, že obálka přesně kopíruje tvar vyobrazeného objektu, zvláště pak u složitějších objektů. Se samotnou fyzikální simulací objektu se zabývá komponenta Rigidbody, která při aplikaci na objekt ovládá komponentu Transform obsahující informace o pozici, rotaci a škálování objektu, a přidává mu další vlastnosti. Objekt se pak více či méně chová jako těleso reálného světa - má určitou hmotnost, působí na něj gravitace, síly, tření, koliduje ostatními objekty s navěšenou komponentou Collider. Vlastnosti komponenty Rigidbody lze upravovat podle potřeb a objekt je pak reflektuje svým chováním ve hře.

2.2.5 Asset store

Příjemnou součástí pro vývojáře v Unity je Asset store, což je knihovna nejrůznějších assetů od ostatních vývojářů, kteří je veřejně publikovali. Tyto assety lze jednoduše stáhnout, importovat do projektu a hned začít používat. Najdeme zde všechno od skriptů, modelů, materiálů, shaderů, až po zvuky, sprity a skyboxy. V určitých případech je tedy možné ušetřit čas a využít již vytvořené assety, které jsou takto volně přístupné.



Obrázek 2.3: Unity Asset Store [1]

■ 2.3 Existující podobné hry

V současné době existuje na obchodě Google Play velké množství her nejrůznějších žánrů. Pro porovnání a ujasnění si vize, jak by přibližně výsledná hra mohla vypadat, jsem vybral pár platformových her, které se mé představě nejvíce přibližují.

■ 2.3.1 Lucid Dream Adventure 3

Platformová adventura s pochmurným příběhem, hráč zde hraje za postavičku Lucy, malé holčičky, která se pohybuje ve své noční můře. Ve svém snu se po cestě potýká s různými hádankami a nástrahami. Vizuální stránka hry je zpracována strašidelným způsobem.



Obrázek 2.4: Lucid Dream Adventure 3 [2]

■ 2.3.2 Typoman mobile

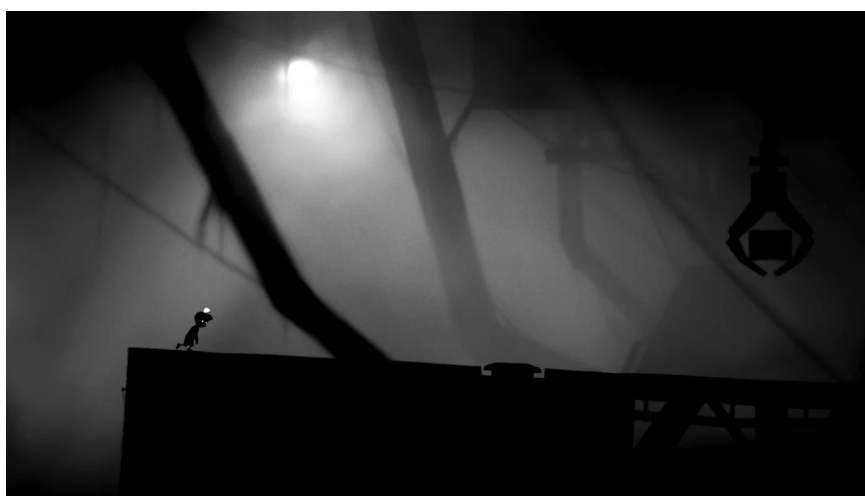
Typoman je puzzle platformová hra, ve které se dostáváte do role hrdiny, který má tu speciální vlastnost, že dokáže tvořit z písmen slova mající vliv na prostředí kolem něj. Pomocí této dovednosti je jeho úkolem projít surrealistický herní svět a vyhnout se smrti.



Obrázek 2.5: Typoman mobile [3]

■ 2.3.3 Limbo

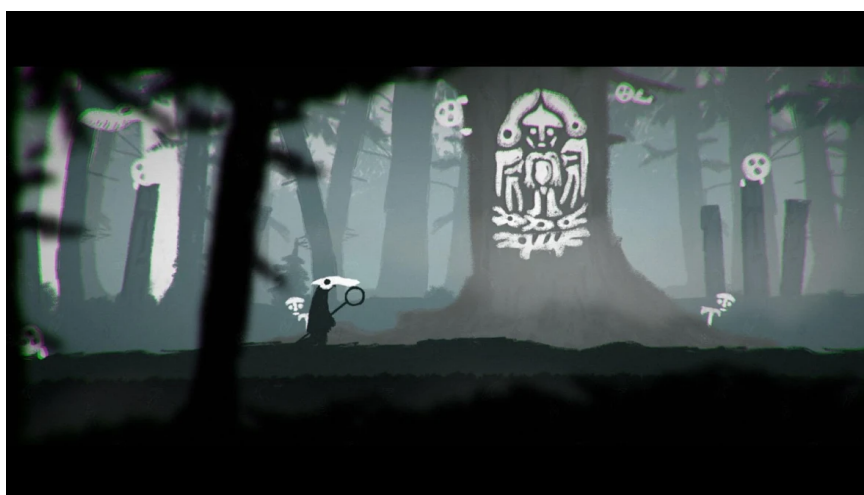
Jedna z nejnámějších her tohoto žánru. Hráč se zde nachází v pozici malého kluka, který se probudí na pokraji lesa. Kluk chce najít jeho zmizelou sestru, a tak se vydává cestou temným lesem skrývajícím nástrahy a nebezpečí, kterým musí hoch čelit. Tato puzzle platformová hra získala několik ocenění a byla po kladném ohlasu portována na další platformy.



Obrázek 2.6: Limbo [4]

2.3.4 The Mooseman

Tato hra nás zavede do mysteriózního světa plného různých neznámých elementů a stvoření. Hrajeme zde za Moosemana, který má dovednost vidět i to, co je lidskému oku neviditelné. Samotná hra je založena spíše na samotném putování, procházení destinací a vyprávění příběhu, najdeme zde ale místy ale i hratelné prvky jako souboj s příšerami či používání obranného štítu proti nadpřirozeným jevům.



Obrázek 2.7: The Mooseman [5]

2.4 Porovnání existujících řešení

Každá výše zmíněná hra má své přednosti - například The Mooseman se na rozdíl od Limbo či Typoman mobile vyznačuje větším důrazem na příběhovou stránku hry, jelikož průchod hrou připomíná spíše vyprávění události než hraní hry. Podobné je to u Lucid Dream Adventure 3, ale hra poskytuje větší vyváženost mezi vyprávěním a samotnou hratelností. V čem se všechny hry shodují, je jejich celkové vizuální zpracování. Ačkoliv každá hra znázorňuje jiné prostředí světa, ve kterém se hráč nachází, všechny hry sdílí společný 2D pohled na scénu, vícevrstvé prostředí okolo hráče, vzbuzující pocit hloubky ve scéně, a stejný cíl hry, tedy dostat se až na konec herního světa. Vizi této práce nejlépe vystihují hry Limbo a Typoman mobile. Je zde kladen mnohem větší důraz na řešení jednotlivých hlavolamů a logických rébusů, což by mělo hráče zaujmout natolik, aby ho hra donutila při hraní místy přemýšlet. V žádném případě tedy nejde o pouhý přímočarý průchod z bodu A do bodu B, nýbrž o hledání cesty, jak takového průchodu bezpečně dosáhnout.

Kapitola 3

Návrh hry

3.1 Herní návrhový dokument

Důležitou součástí vývoje hry je Herní návrhový dokument. Tento dokument slouží jako vize, jak bude výsledná hra vypadat a jaké budou její náležitosti. Zároveň poskytuje jakýsi organizační nástroj pro všechny, kteří se podílí na vývoji hry [15]. V obsahu dokumentu najdeme informace o příběhu, postavách, prostředí, ozvučení, ovládání, a případných dalších částí hry. Výhody vytvoření herního návrhového dokumentu se projeví především při vývoji ve větším týmu, rozhodně ale není na škodu zhotovit dokument i při vývoji jako jednotlivec - díky tomu získáme lepší přehled a vizi o cíli, kterého chceme dosáhnout, a ujasní se vše, co k tomu bude potřeba.

3.1.1 Koncept

Hra nese název Forgotten Forest a jedná se o platformovou hru, která se odehrává v lese. Ve hře je cílem dostat se s hráčovou postavou přes temný les, který skrývá různé překážky, nebezpečné nástrahy (padající předměty, propasti, ...). Po cestě je třeba plnit různé úkoly (posouvání krabic, tahání páček, ...), aby hráč mohl pokračovat v cestě dál a vyhnul se nástrahám a nebezpečí. Hráčův úkol je projít zapomenutým lesem a dostat se zpět domů.

■ 3.1.2 Příběh

„Ach, moje hlava... co se stalo, kde to jsem? Nevzpomínám si, že bych šel do lesa.. vlastně si ani nepamatuji, kde jsem naposledy... moment, tento les poznávám! Po této cestě chodím zpět domů z práce... ale co tu dělám teď, když je venku tak temno a už se stmívá? Běžně tudy chodím domů ještě za světla... no nic, snad si vzpomenu později. Teď se musím dostat domů.“

■ 3.1.3 Postavy

- **Panáček** – postava kterou hráč ovládá a okolo které se celé dění odehrává. Umí chodit a skákat, plnit jednoduché úkony, jako například tlačit bednu, tahat páčky a tak podobně.
- **Lučištník** – neznámý člověk s lukem a šípy, který střílí po Panáčkovi. Zasažení Panáčka šípem znamená jistou smrt. Pokud se hráč dostane blízko k Lučištníkovi, Lučištník se dá na útěk.
- **Pavouk** – nebezpečný tvor, který v dostatečné blízkosti hráče Panáčka zabije. Jedinou obranou proti němu je útěk.

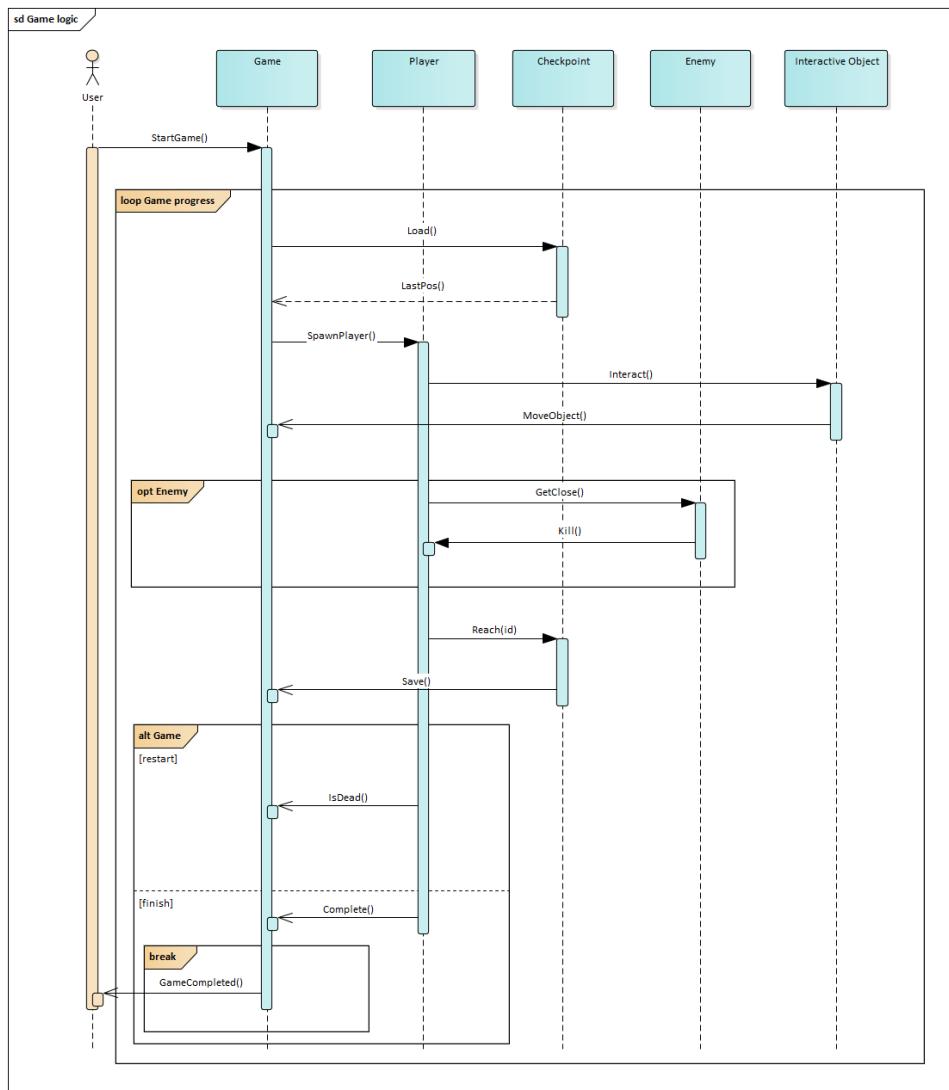
■ 3.1.4 Level design

Herní svět je navržen tak, aby sám o sobě působil jako jeden celek. Není zde více úrovní, nýbrž pouze jedna velká úroveň, která v sobě ukrývá veškerý obsah hry. Kamera na scénu je přiblížená tak, že hráč vždy vidí pouze blízké okolí kolem svého panáčka - nikdy tedy nevidí větší část světa, než tu, kde právě je. Při postupu hráče hrou se obtížnost jednotlivých překážek a hádanek zvyšuje, roste i počet nepřátel, se kterými se hráč setká. Les také obsahuje například kutálející se balvany či padací předměty, kterým je třeba se vyhnout. Jednou za čas hráč narazí na jeskyni, vodu, či propast, přičemž bude jeho úkolem vymyslet, jak se přes takové překážky dostat.

■ 3.1.5 Hratelnost

Hráč má k dispozici základní úkony pro ovládání postavy, jako je pohyb doleva a doprava, skok, interakci s určitými předměty. Ovládací prvky pro

tyto úkony budou permanentně zobrazeny na obrazovce hry. Dále hra bude automaticky ukládat hráčův postup. V případě, že hráč umře, bude oživen v posledním dosaženém checkpointu. Tyto checkpointy nejsou pro hráče viditelné, typicky ale následují po vyřešení úkolů či překonání nebezpečné nástrahy. Proti nepřítelům nemá hráč jinou možnost se bránit, než se schovat, uhnout, nebo utéct, jelikož ve hře nebude možnost přímého boje ze strany hráče.



Obrázek 3.1: Sekvenční diagram logiky ve hře

Kamera ve scéně, která má za úkol zprostředkovávat zobrazování dění ve hře hráči, bude nastavena do ortografického režimu. Toto nastavení vyvolá dojem 2D prostředí, ačkoliv je scéna ve skutečnosti 3D. Jedním z důvodů, proč je zde vhodné použít tento typ projekce, je neměnné zobrazování terénu při výskoku hráče. Při perspektivní projekci je totiž při záběru kamery nad úrovní země vidět, že zem je ve skutečnosti, zjednodušeně řečeno, jen úzký

kvádr bez větší hloubky směrem od hráče.

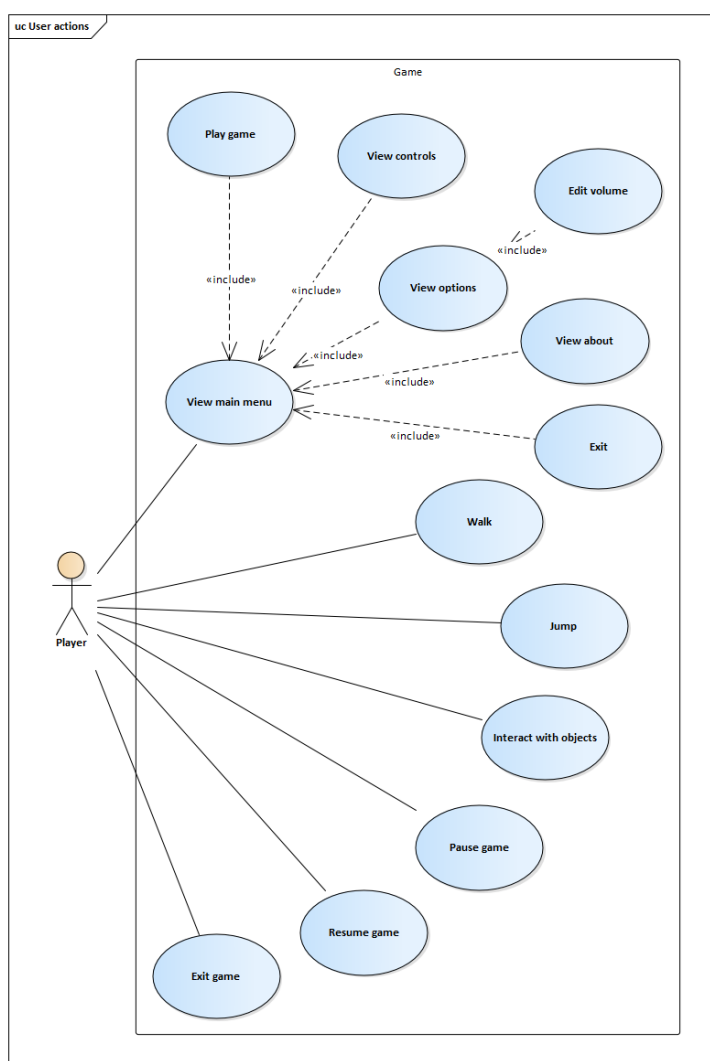
■ 3.1.6 Hudba a zvuky

Hra obsahuje temnou ambientní skladbu, která má za úkol vcítit hráče do situace Panáčka. Hudba je klidnějšího rázu, vyvolávající nejistotu, a dostatečně dlouhá na to, aby mohla běžet ve smyčce po celou dobu hraní. Konkrétní skladbu hudby je vhodné vybrat co nejdříve, abychom už od počátků práce na hře ujasnili, jakou atmosféru bude naše hra vyzařovat [16]. Zvuky a efekty jsou autentické pro jednotlivé akce - pro chození je použit zvuk šustění listí, pro valící se balvan zvuky šutrů a otřesu země, a tak podobně.

■ 3.1.7 Ovládání

Veškeré ovládání je zobrazeno na obrazovce hry a skrze něj hráč ovládá svého Panáčka. S hrdinou lze provádět tyto úkony:

- pohybovat se panáčkem vlevo a vpravo
- vyskočit a pokrčit se
- interagovat s blízkými předměty, pokud to dovolují



Obrázek 3.2: Use case diagram hráče

3.1.8 Uživatelské rozhraní

- **Hlavní menu** – zobrazí se po startu hry. Obsahuje tyto tlačítka:
 - **Hrát** – spustí hru od začátku, nebo tam, kde naposled hráč skončil (pokud již hru hrál)
 - **Nastavení** – ukáže panely hlasitosti pro hudbu a zvuky, které lze individuálně nastavit
 - **Ovládání** – zobrazí popis ovládání a rozhraní ve hře
 - **O hře** – zobrazí obrazovku s informacemi o hře
 - **Konec** – ukončí hru

- **Hra** – obrazovka přímo ve hře, obsahující tyto ovládací prvky:
 - **Pauza** – zastaví hru a zobrazí menu s tlačítky:
 - **Pokračovat** – obnoví hru
 - **Poslední záchytný bod** – hráče umístí na poslední záchytný bod
 - **Menu** – vrátí hráče do menu
 - **Šipka vlevo** – pohyb Panáčka doleva
 - **Šipka vpravo** – pohyb Panáčka doprava
 - **Šipka nahoru** – výskok Panáčka
 - **Šipka dolů** – pokrčení Panáčka
 - **Použit** – Panáček bude interagovat s předmětem, pokud je nějaký v jeho blízkosti

■ 3.2 První spuštění hry

Po zobrazení splashscreenu s logem Unity se běžně zobrazí hlavní menu hry. Hráč má možnost projít si ovládání a nastavení hry. Tlačítkem *Hrát* se spustí herní scéna, přičemž se přes ní zobrazí s poloprůhledným pozadím textově psaný příběh. Po dotknutí se obrazovky příběh zmizí, hráč se ocitá ve hře a může hrát.

■ 3.3 Funkční a nefunkční požadavky

Funkční požadavky

- po prvotním spuštění hry se před začátkem hraní zobrazí příběh
- uživatel může nastavovat hlasitost hudby a zvuků
- zpětné tlačítko systému Android bude ve hře implementováno
- hráč může interagovat s určitými předměty
- hráč se může pohybovat a skákat

Nefunkční požadavky

- hra musí plnohodnotně fungovat na systému minimálně Android 4.4 a vyšší
- hra musí stabilně běžet na alespoň 30 snímků za sekundu
- hra bude kompletně v češtině
- ovládání musí být zřejmé a jednoznačné
- hra se musí správně zobrazovat na různých velikostech displejů
- Instalační balíček .apk nepřesáhne 100MB

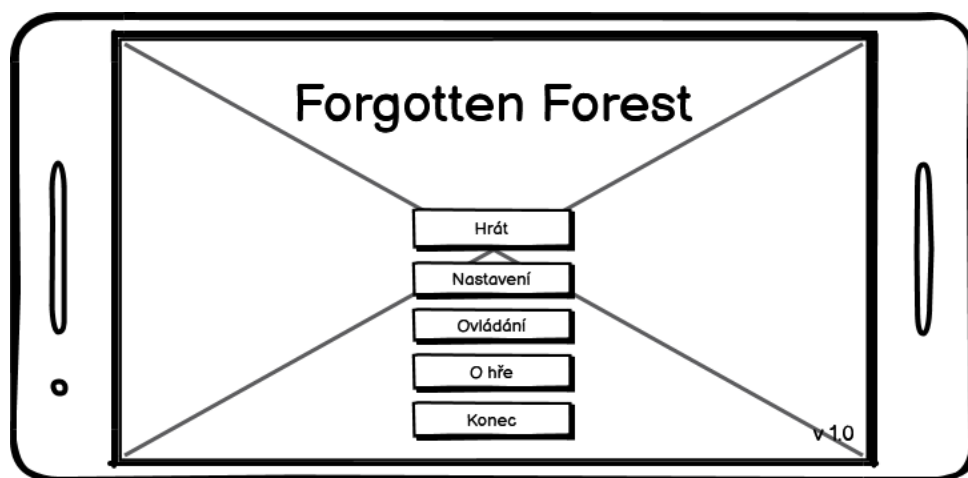
3.4 Uživatelské rozhraní

Aby byla mobilní hra svým uživatelským rozhraním pro hráče přívětivá, je třeba správně rozložit její ovládací prvky po displeji. Na obrázku níže lze vidět zelené oblasti, kde je displej dobře přístupný, a tudíž je vhodné umístit ovládací elementy někam v této oblasti. V případě nutnosti lze ovládací prvky umístit i do oranžové zóny, je ale lepší se tomu kvůli přístupnosti a celkovému požitku ze hry vyvarovat.

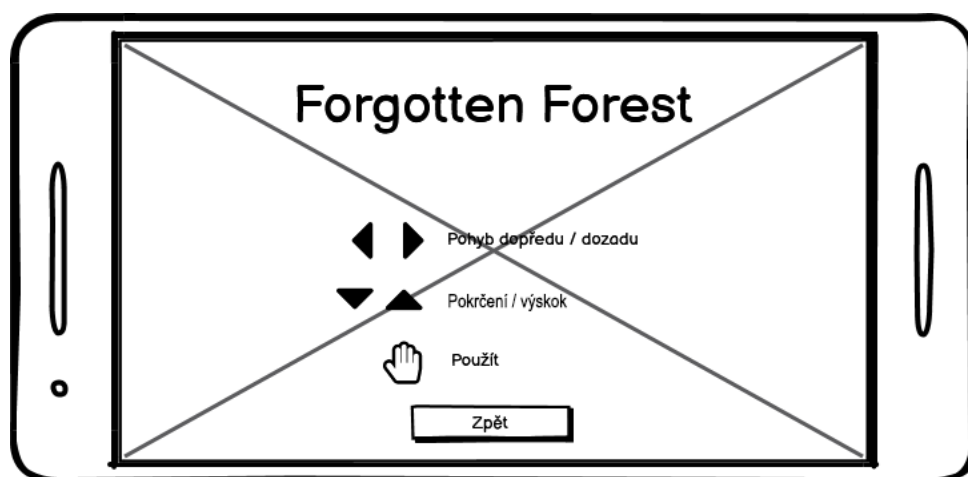


Obrázek 3.3: Dostupnost dotykové obrazovky v landscape režimu [6]

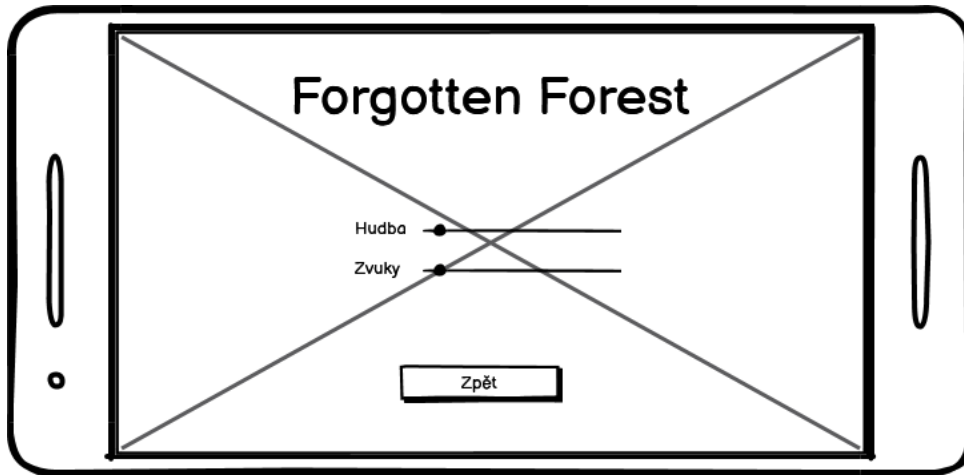
Na obrázcích níže jsou zobrazeny low-fidelity obrazovky uživatelského rozhraní, které se ve hře nachází.



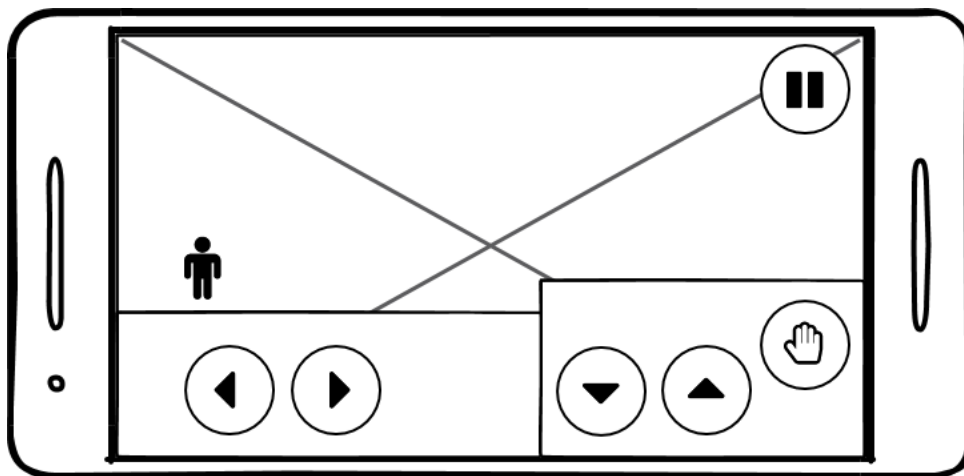
Obrázek 3.4: Hlavní menu



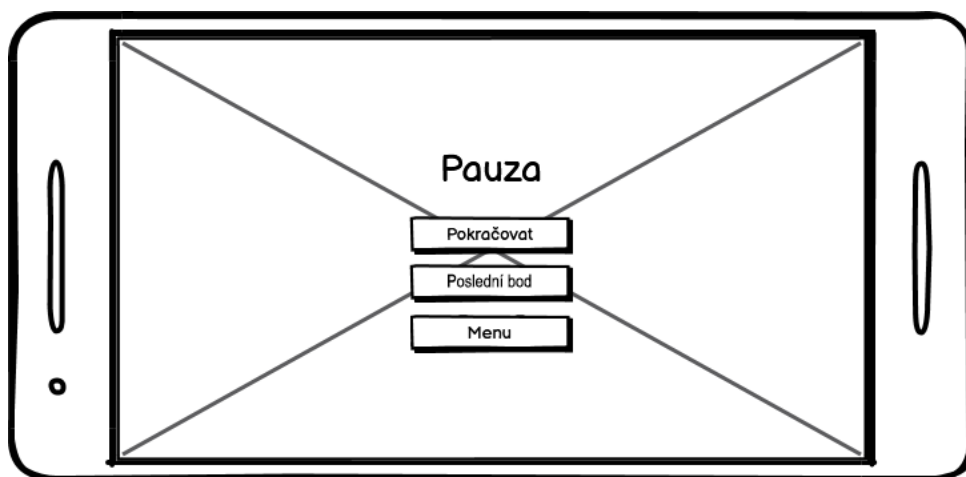
Obrázek 3.5: Ovládání



Obrázek 3.6: Nastavení



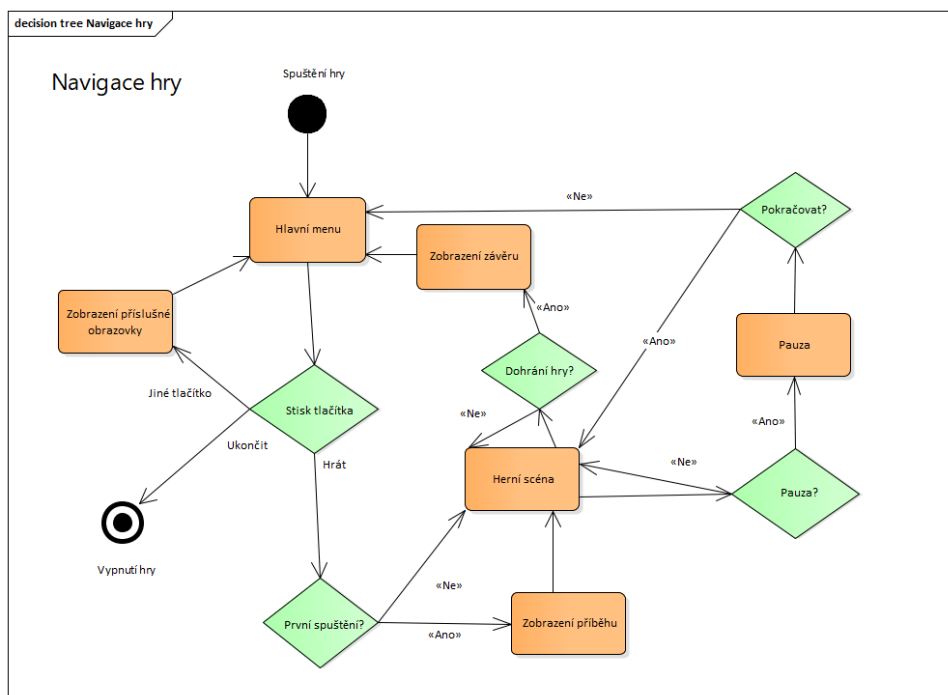
Obrázek 3.7: Herní scéna



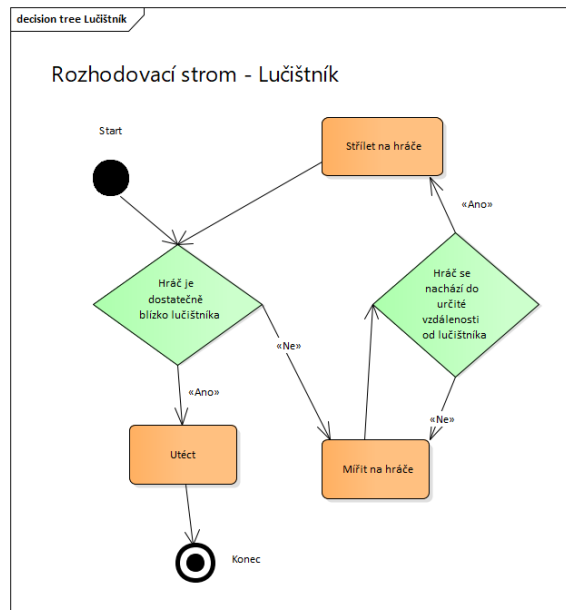
Obrázek 3.8: Pauza

3.5 Diagramy hry

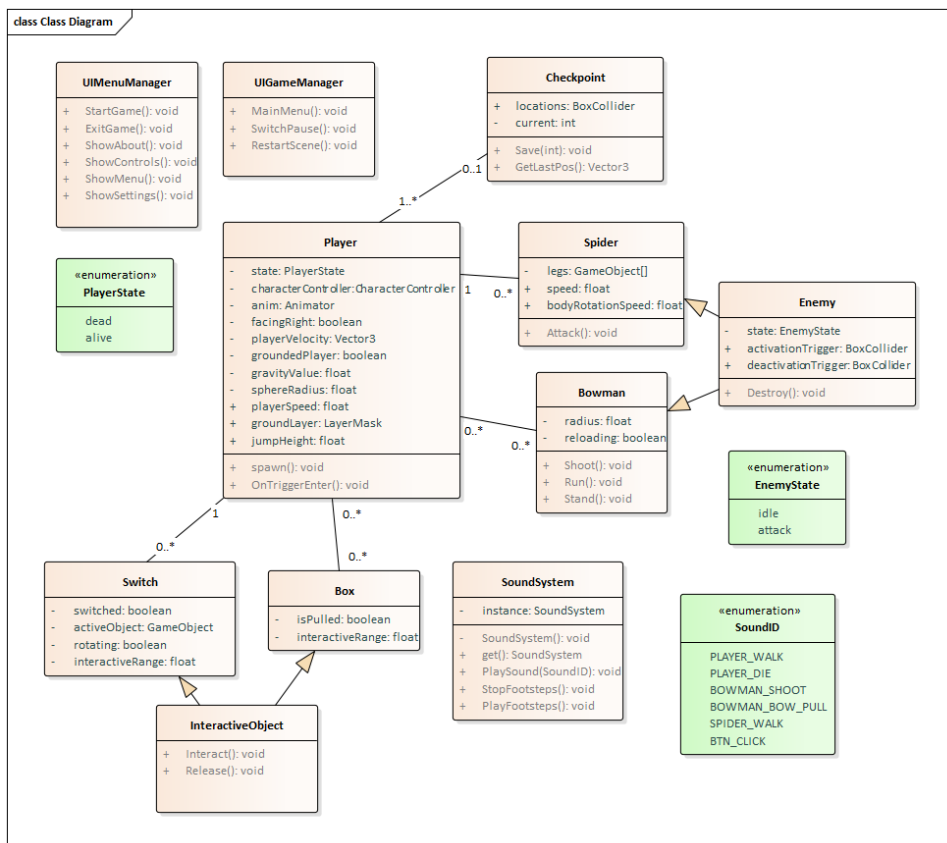
Pro ujasnění úkonů a činností v jednotlivých částech hry byly vytvořeny diagramy navigace hry a chování postavy Lučištník. Dále byl vyhotoven diagram tříd pro významnější třídy v projektu.



Obrázek 3.9: Navigace obrazovek ve hře



Obrázek 3.10: Rozhodovací diagram postavy Lučičník

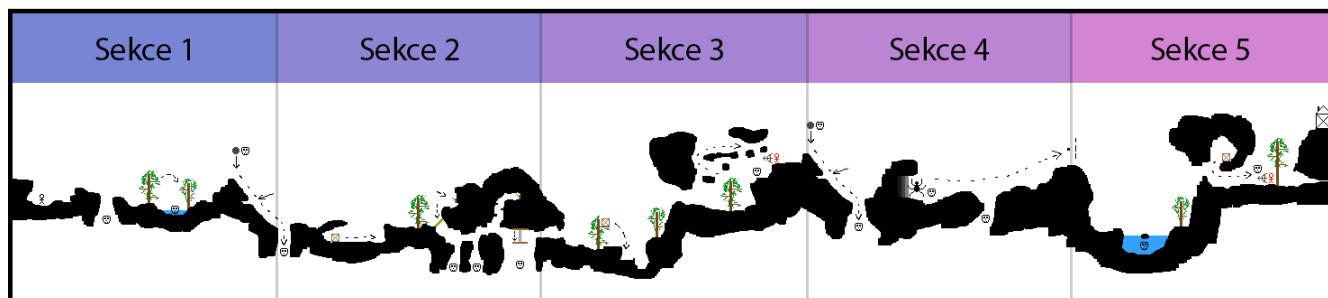


Obrázek 3.11: Diagram významnějších tříd ve hře

3.6 Části herního světa

Z hlediska návrhu jsem herní svět pomyslně rozdělil do pěti sekcí, kde každá sekce přináší něco nového oproti části předchozí. Počet sekcí byl vybrán s ohledem na rozsah celkové hry a počet překážek a úkolů, které se v ní nachází.

Herní svět



Obrázek 3.12: Vyobrazení herního světa

- **Sekce 1** – první část slouží k seznámení hráče s mechanikou hry. Vyskytují se zde jednoduché překážky jako kopce, voda, či propasti, které je třeba přeskočit.
- **Sekce 2** – druhá část obsahuje převážně nástrahy v podobě padajícího balvanu, propastí, a menší jeskyní se zavěšenou plošinou. V druhé sekci se nachází i část, kde hráč poprvé interaguje s krabicí.
- **Sekce 3** – ve třetí části se hráč poprvé setká s postavou Lučištníka, který po něm střílí z luku. K přežití musí uhýbat a skrývat se za objekty.
- **Sekce 4** – čtvrtá část skrývá nebezpečného Pavouka, který přileze z levého okraje obrazovky a pronásleduje hráče, který mu musí po určité době utíkat.
- **Sekce 5** – v poslední sekci hráč narazí na dalšího lučištníka a nakonec konečně dorazí ke svému domu.

Jednotlivé sekce mohou obsahovat různé části i z již předchozích sekcí.



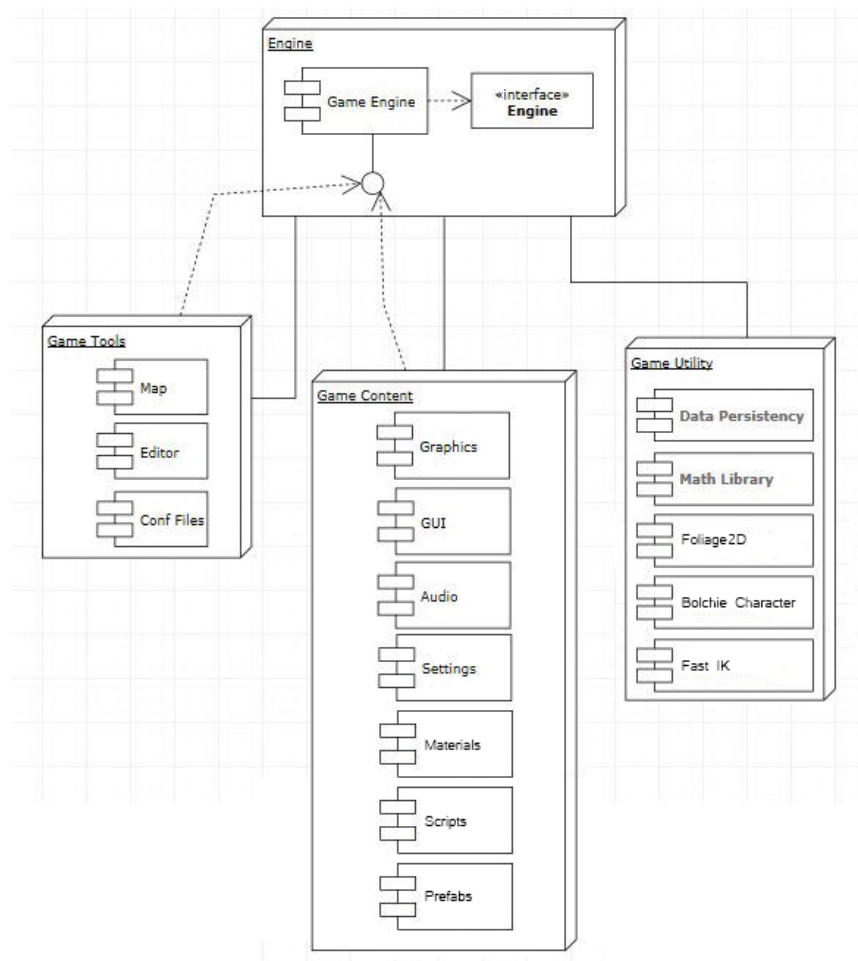
Kapitola 4

Implementace



4.1 Architektura

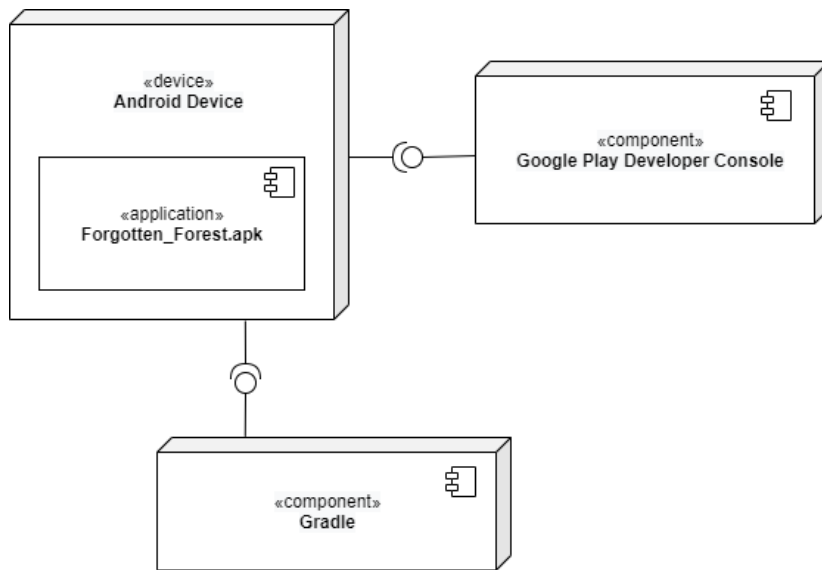
Důležitou součástí pro vývoj hry je samotný herní engine. Jeho vliv sahá od stanovení struktury a organizace souborů v projektu, práci se soubory, zaznamenávání vstupů, processing a vykreslování, až po výslednou kompilaci a exportu výsledného programu. Na obrázku níže je vyobrazen component diagram herního engine projektu.



Obrázek 4.1: Component diagram herního engine projektu [7]

Jednotlivé komponenty Game Tools, Game Content a Game Utility jsou nedílnou součástí herního engine. Game Tools odkazují na všechny nástroje používané k vývoji hry, jako editaci scény, postav, konfiguračních souborů. Game content obsahuje všechny zdroje použité ve hře, jako je grafika, zvuky, skripty, materiály a další. V komponentě Game Utility získáváme propojení mezi Game Tools a Game Content, najdeme zde i externí knihovny [7].

Unity při buildu hry využívá Gradle, nástroj pro automatizaci sestavení programu. Po exportu hry z herního engine získáme výsledný soubor s hrou pro platformu Android s koncovkou .apk. Tento balíček apk lze následně nainstalovat na kterémkoliv Android zařízení (tato aplikace vyžaduje ke spuštění instalace alespoň Android verze 4.4). Pro vystavení hry na Google Play je třeba nahrát instalační balíček do Google Play Developer Console, přes kterou je možné hru nadále spravovat a distribuovat.



Obrázek 4.2: Component diagram distribuce hry

4.2 Technologie

Jak již bylo zmíněno v sekci návrhu hry, pro realizaci tohoto projektu jsou využity komponenty, které jsou v herním engine již dostupné. V následujících sekcích je popsáno, pomocí kterých těchto komponent realizovány určité části hry.

4.2.1 Uživatelské rozhraní

Zobrazování prvků rozhraní je realizováno pomocí Canvasu, který si lze představit jako plátno, na které můžeme umísťovat uživatelské rozhraní. Všechny prvky, jako jsou například Buttons, Toggles, Images, Sprites, či Sliders jsou zobrazovány “přes” herní scénu. Když jsou aktivovány, jsou pro hráče viditelné a interaktivní. Toho využijeme ve hře, když hráč klikne na tlačítko “Pauza”, aktivuje a zobrazí se příslušné CanvasGroup (skupina určitých prvků rozhraní) obsahující obrazovku s pauzou a ovládacími prvky. Při kliknutí na “Pokračovat ve hře” opět tento CanvasGroup deaktivujeme a hráč vidí pouze jen uživatelské prvky určené pro herní akci.

Jednotlivé uživatelské prvky lze přichytit na určitá místa na obrazovce - například doprostřed, doprava dolů, nebo vlevo nahoře. Díky tomu lze zařídit relativní konzistentnost pozice prvků na různých displejích a zařízeních.

Většina tlačítek ve hře bude takto ukotvena, aby se provádělo jejich škálování obdobně na všech zařízeních.

Aby bylo ovládání pro uživatele přívětivější, tlačítka jsou při stisknutí/podržení animovaná tmavě šedou barvou (využijeme zde naimplementované funkcionality Buttonu, který nám toto chování nabízí).

■ 4.2.2 Pohyb postav

Pro pohyb a animování postav využijeme návrhového vzoru State, kdy se postava chová či vypadá podle toho, jaký má zrovna interní stav. Díky tomuto vzoru zamezíme chování postavy, které nechceme aby nastalo - například provést výskok, když je Panáček již ve vzduchu. Vzor totiž spočívá v definování jasných přechodů z jednotlivých stavů, které mohou nastat [17]. V případě, že nevytvoříme mezi dvěma stavy žádný přechod, nebude možné napřímo mezi těmito stavy přepínat.

Na obdobném principu funguje i Unity integrovaná komponenta Animator Controller, která slouží k uspořádání a nastavení různých animačních klipů a přechodů pro danou postavu. Jak již název napovídá, pomocí této komponenty ovládáme animace jednotlivých postav, založené na jejich aktuálních stavech. Mezi ně řadíme například chůzi, výskok, natahování luku, interakci s předměty. Výhodou Animator Controlleru je, že mezi jednotlivými animačními klipy lze postupně prolínat - jednotlivé klipy se pak mezi sebou míchají podle toho, jaká je momentální hodnota tohoto přechodu. To přináší značnou výhodu v tom, že při změně stavu postavy je její animace pro hráče vypadat plynule a nedojde k žádným animačním záškubům.

■ 4.2.3 Prostředí

K vymodelování terénu je možné využít přídatný asset Probuilder vydaný přímo od Unity technologies. Pomocí tohoto nástroje je možné interaktivně modelovat objekty přímo v editoru, bez potřeby používat modelovací software třetích stran. Ušetří se tak čas, kdy by se terén modeloval v externím modelovacím softwaru a následně převáděl do Unity. Probuilder má k dispozici základní funkcionality pro úpravu objektů, které pro vytvoření terénu budou více než dostačující. Jelikož je ovšem tato hra vyvíjena na mobilní zařízení, je třeba si dát pozor a nevytvářet až příliš detailní terén, s ohledem na hardwarově slabší zařízení.

Pro kutálející se balvanu po povrchu, kterým se hráč musí vyhnout, je vhodné navrhnout vlastní cestu, po které se balvan bude kutálet - tím zaručíme deterministického chování balvanu pro každou instanci hry. Tato cesta se skládá z jednotlivých bodů a ty by měly být umístěny tak, aby kutálení balvanu a jeho fiktivní odražení od povrchu působilo co nejrealističtěji. Hráč tuto cestu a její body samozřejmě neuvidí.

Pro herní element voda se nabízí využít obrázek či barvu s modrým zbarvením. Není žádoucí vytvářet či užívat jakoukoliv pokročilejší simulaci vody, s ohledem na možné omezení hardware uživatelských zařízení. Voda se nachází ve hře pouze ojediněle a jedná se o překážku, kterou hráč (Panáček) musí překonat vzduchem, aby se neutopil.

Samotné pozadí hry je rozostřené, pomalu se posunující při pohybu hráče, aby vyvolalo dojem hloubky ve scéně. Toto pozadí je implementováno jako nekonečně posouvající se pozadí - vedle sebe budou dva totožné obrázky, přičemž když jeden obrázek zmizí z obrazovky (hráč urazí určitou vzdálenost), umístí se mimo obrazovku před momentální obrázek na obrazovce. Pomocí tohoto triku hráč nic nepozná a pozadí se pro něj jeví jako nekonečný průchod lesem.

■ 4.2.4 Interakce s předměty

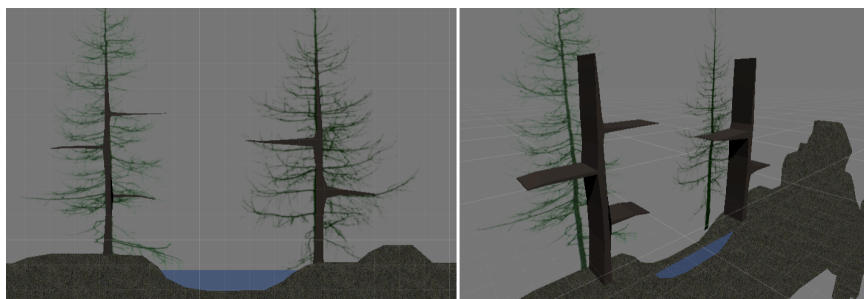
Panáček může interagovat s určitými předměty, jako jsou například bedny. Když hráč stiskne tlačítko interakce, zjistí se vzdálenost nejbližšího předmětu k hráči. Pokud je dostatečně malá, spustí se animace Panáčka jak natáhne ruce k danému předmětu.

■ 4.3 Herní svět

V této sekci je popsána implementace a způsob řešení některých objektů a částí vyskytujících se ve hře.

4.3.1 Stromy

Většina stromů se na mapě jeví jako stromy v pozadí. Jedná se o statické obrázky, umístěné ve vrstvě objektů nacházejících se za postavou hráče. Na některé stromy ovšem může hráč vylézt. Pomocí objektů kopírující vybrané větve stromu je docíleno možného skákání po větvích stromu na obrázku. To, že na určitý strom lze vylézt, lze poznat pomocí znatelně silnějších větví, které vykreslují právě tyto pomocné objekty.



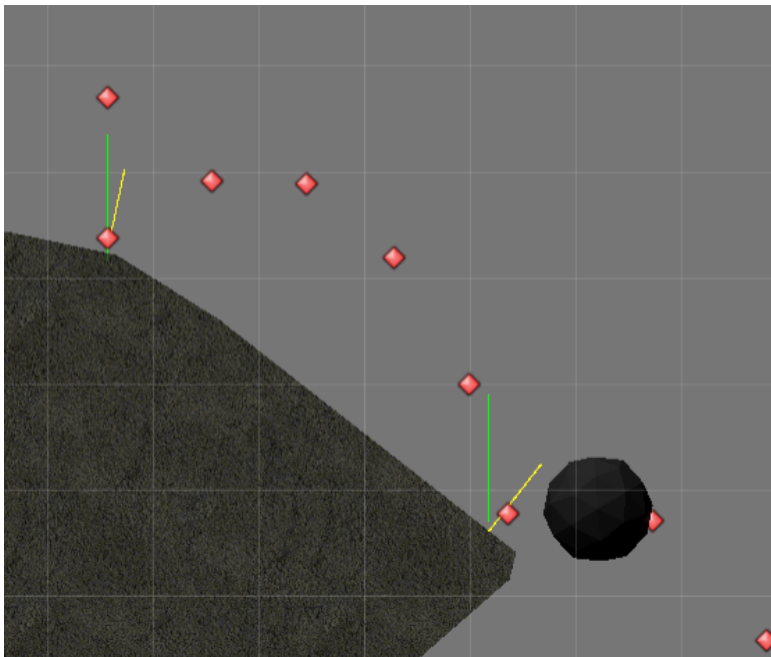
Obrázek 4.3: Stromy na které lze vylézt, pohled ze hry (vlevo) a ve scéně (vpravo)

Tyto objekty mají zvláštní vlastnost, že hráč jimi může proskočit zezdola, ale z jakékoliv jiné strany s nimi bude kolidovat. Tohoto chování je docíleno dynamickým aktivováním a deaktivováním kolizních komponent pomocných objektů dle aktuální rychlosti a směru pohybu hráče. Zjednodušeně řečeno, když se hráč pohybuje směrem vzhůru, objekty se vypnou, v opačném případě zůstávají zapnuté. Na větve lze tak naskočit zezdola a zůstat na nich stát.

4.3.2 Balvan

Tento objekt hráče při prvním setkání zřejmě překvapí. Při vstupu Panáčka do určité oblasti na mapě se aktivuje skript balvanu, který je umístěn kousek zpětně od hráče. Balvan následně padá a kutálí se ze svahu, kde v případě zasáhnutí hráče postava Panáčka umírá.

Ačkoliv se může zdát, že kutálení balvanu je zde proces fyzikální simulace hry, jedná se ve skutečnosti o skriptem řízený objekt po cestě bodů předem vytvořené cesty. Důvod, proč nebyla využita fyzikální simulace dostupná přímo v Unity, je popsán v sekci 4.2.3 - Prostředí.



Obrázek 4.4: Vyobrazení kutálejícího se balvanu (ladění)

Na obrázku je vidět cesta tvořená z červených bodů, kterou balvan při pádu následuje. Při kontaktu balvanu se zemí je přehrán zvuk dopadu kamene. Pro rotaci kamene při jeho odražení od země zjistíme v daném místě normálový vektor terénu. Ten je na obrázku vyznačen žlutou barvou. Kámen na mapě padá vždy z kopce směrem doprava dolů, takže se zde nabízí možnost porovnat sevřený úhel vektoru normály a vektoru směřující vzhůru (zelený vektor). Při získání úhlu je toto číslo vynásobené konstantou a výsledek je přičten k celkovému aktuálnímu proměnnému číslu reprezentující přídavnou rotaci. Při každém snímku je pak tato rotace připočtena k aktuální rotaci objektu. Kámen tak ve hře ve výsledku vypadá, že se skutečně kutálí po zemi.

■ 4.3.3 Bedna

Tlačení a tahání bedny je vyřešeno hierarchickým způsobem skládání objektů ve scéně. Když se hráč nachází v blízkosti bedny a stiskne tlačítko interakce, může s bednou pohybovat po mapě. Při zahájení interakce se krabice v hierarchii scény umístí jako potomek (child) postavy hráče (parent). Při každém pohybu hráče tak krabice kopíruje pozici postavy, která jí tlačí. Při uvolnění interakce se krabice vrátí zpět na původní místo ve scéně, tedy již nemá hráče jako rodiče a nekopíruje jeho pohyb.

Jelikož je tento úkon tlačení bedny řešený tímto způsobem, je zde přidáno i několik podmínek, za kterých je možné s krabicí interagovat, aby ve hře nevznikaly příliš nereálné situace. Podmínky pro možnou manipulaci s bednou jsou:

- hráč je dostatečně blízko krabice
- pozice hráče v ose Y je v rozmezí horního a spodního bodu krabice
- krabice se nepřekrývá s okolním terénem
- hráč je otočen směrem ke krabici
- hráč je na zemi

Pokud jakákoliv z těchto podmínek není splněna, automaticky je jakákoliv interakce zrušena bez ohledu na hráčem stisknuté tlačítko. Pro zahájení interakce musí hráč stisknout tlačítko znovu.

■ 4.3.4 Páčka

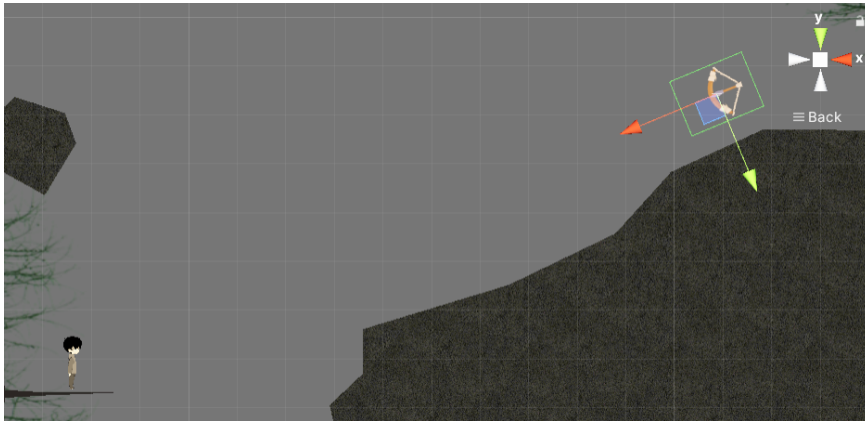
Páčka je jednoduchý objekt aktivující jiné objekty ve scéně. Každá páčka lze zatáhnout pouze jednou. Složena je z pevné nehybné podstavy a pákou, která má ukotvení v místě, kde je do pevné podstavy zasunuta. Při interakci se páka začne určitou rychlostí otáčet směrem dolů. Když toto rotování urazí od počátečního natočení 60 stupňů, rotace se zastaví a páka je úspěšně zatažena.

■ 4.3.5 Lučištník

Lučištník je postava, která míří lukem na Panáčka a čas od času na něj vystřelí šíp. Vystřelený šíp letí lineárně po přímce, která je rovnoběžná s lokální osou X luku v momentě výstřelu. Šíp je zničen po nárazu do prvního objektu - země, krabice, či jiného předmětu. V případě zasáhnutí hráče postava Panáčka umírá.

Aby luk v reálném čase mířil na hráče, je provedeno několik výpočtů. Nejdříve odečteme pozici luku od aktuální pozice hráče. Získáme vektor udávající směr, kterým chceme luk natočit. Posléze zjistíme úhel mezi osou X

a získaného vektoru. O velikosti tohoto úhlu vytvoříme Quaternion (struktura v Unity používaná pro reprezentaci rotací) rotující okolo osy Z (hloubka). Nakonec přiřadíme luku jako rotaci tento Quaternion. Luk bude nyní natočen na hráče.

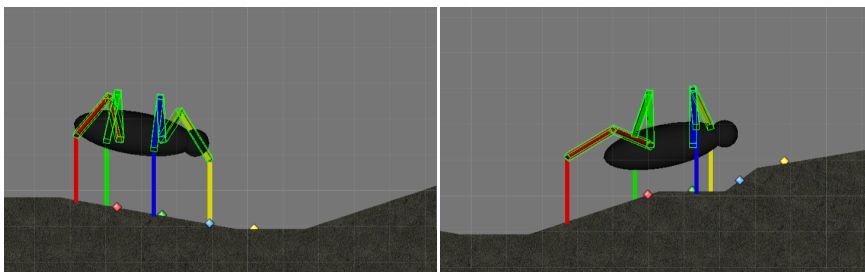


Obrázek 4.5: Luk mířící na Panáčka (ladění)

4.3.6 Pavouk

Pavouk je strašidelným nepřítelem, který se zjeví z kraje obrazovky a pronásleduje hráče tak dlouho, dokud mu hráč neuteče.

Narozdíl od balvanu, Pavouk nemá konkrétní cestu vytvořenou z jednotlivých bodů. Pohyb Pavouka, respektive jeho noh, je realizován pomocí inverzní kinematiky a ray castingu z pozic jednotlivých bodů směrem na zem. Inverzní kinematika je proces, který hledá pro daný koncový bod hodnoty jednotlivých kloubů více na sebe navazujících těles tak, aby ve výsledku těleso na koncový bod dosáhlo. Ray casting je vyslání paprsku z určité pozice zvoleným směrem, a následné zjištění různých informací při jeho nárazu.



Obrázek 4.6: Pochod Pavouka (ladění)

Na obrázku výše můžeme vidět Pavouka ze hry, který má pro ilustraci každou nohu jinak barevnou. Každá noha má svůj vlastní cílový bod, označený

totožnou barvou. Tyto body se při aktivaci Pavouka pohybují konstantně vpřed a kopírují povrch terénu, čehož je dosaženo pomocí ray castingu. Pro každou nohu se následně vyhodnocuje, zda je její vzdálenost od jejího cílového bodu příliš velká. V případě, že ano, přemístí se konec nohy na aktuální místo tohoto bodu. Inverzní kinematika se při tomto přechodu postará o správné natočení a pozicování předchozích částí nohy, jelikož koncové body pro inverzní kinematiku kopírují pozice vrcholů posledních částí pavoučích noh. Celý tento proces se pak opakuje pro každou nohu Pavouka zvlášť. Ve výsledku tak tento Pavouk dokáže lézt i po více převýšeném terénu či strmém svahu.

Samotné tělo Pavouka se pohybuje stejnou rychlostí jako cílové body jednotlivých noh. Rotace těla se vyhodnocuje podle rozdílu Y pozic pravých a levých noh a jejich následného součtu znásobeného konstantou.

Kapitola 5

Testování

Hra byla před vydáním podrobena uživatelským testům, které měly za úkol odhalit potenciaální problémy a chyby ve hře. Testování probíhalo způsobem stejným, jako kdyby koncový uživatel poprvé stáhl a hrál tuto hru. Úkolem všech testerů bylo zahrát si hru od začátku do konce, nahlásit chyby se kterými se setkali a navrhnout případné úpravy herních prvků, od kterých očekávali jiné chování.

5.1 Provedení uživatelských testů

Testování se zúčastnilo pět uchazečů. Tuto skupinu tvořili muži ve věku 20 až 55 let s mobilním zařízením Android, přičemž dva účastníci nejsou aktivními hráči mobilních her.

Některé připomínky od jednotlivých testerů byly shodné. Při testování nebyla nalezena žádná kritická chyba, která by zásadně ovlivnila běh či průchod hry (způsobem, že by hráč nemohl pokračovat v hraní dál). Níže je popsána zpětná vazba získaná od testerů.

-
- **Poznámka** – při skákání po stromě hráč naráží hlavou do větví nad

ním a po stromě se tak špatně leze.

- **Popis problematiky** – větve některých stromů jsou tvořeny z neviditelných objektů, díky kterým po nich lze chodit. Větve jsou často nad sebou a hráč tak při výskoku může narazit do větve nad ním.
- **Vyhodnocení** – skákání po větvích není takto příliš intuitivní. Neviditelné objekty byly upraveny tak, že hráč nimi může proskočit, pokud nimi prochází zdola nahoru.

-
- **Poznámka** – při zobrazení úvodního příběhu je na obrazovce část textu, která se ztrácí v místě výřezu zařízení.
 - **Popis problematiky** – text se zobrazuje přes celou šířku zařízení. V případě zařízení s výřezem v displeji se může stát, že některé části textu budou zakryty tímto výřezem.
 - **Vyhodnocení** – pro úvodní a závěrečný text bylo nastaveno větší odsazení od okraje obrazovky.

-
- **Poznámka** – při stání na vstupní bráně u vstupu do jeskyně má Panáček animaci jako když padá dolů.
 - **Popis problematiky** – přepnutí stavu animace Panáčka na pád je řízeno skriptem, který kontroluje vzdálenost hráče od země a jeho vertikální sílu působící na něj. Vstupní brána neměla nastavenou stejnou vrstvu jako zem, takže se při vyhodnocování vzdálenosti od země nepočítala.
 - **Vyhodnocení** – pro bránu byla nastavena správná vrstva a Panáček tak už nebude zobrazovat animaci pádu při stání na ní.

-
- **Poznámka** – hráč neočekával, že se ve vodě utopí.

- **Popis problematiky** – panáček se při namočení do vody utopí, protože neumí plavat. Nikde není ovšem řečeno, zda do vody smí nebo ne.
- **Vyhodnocení** – před vodou byl prozatímne umístěn záchytný bod. Do budoucna je vhodné zvážit vysvětlení některých herních elementů hráčovi přímo ve hře, při prvním setkání s nimi.

-
- **Poznámka** – zavěšená plošina v jeskyni se občas začne nereálně pohybovat.
 - **Popis problematiky** – plošina je v jeskyni zavěšená na řetězu objektů využívající Hinge Joint, což je komponenta pro propojení objektů. Nastavená hmotnost těchto objektů byla nastavena na příliš malé číslo, takže fyzikální vlastnosti plošiny nepůsobily vůbec reálně.
 - **Vyhodnocení** – hmotnost spojujících objektů byla navýšena a plošina už by tak neměla svým chováním působit nereálně.

-
- **Poznámka** – v určitých místech bylo těžké doskočit z jednoho místa na druhé.
 - **Popis problematiky** – některé propasti vyžadují poměrně přesný skok z téměř úplného kraje propasti v místě odrazu, aby hráč úspěšně přeskočil propast.
 - **Vyhodnocení** – nejvíce obtížné propasti byly zúženy, aby přeskok na druhý konec nebyl tak náročný.

■ 5.2 Vliv zpětné vazby na hru

Po vyhodnocení uživatelských testů byly nalezené chyby opraveny a zmíněné části hry upraveny. Žádný z účastníků nezaznamenal větší chyby, které by vyžadovaly přepracování větší části hry.

Po zahrání hry testery byly vzneseny některé návrhy na další vylepšení hry, jako je přidání další grafiky, implementace více zvuků, úprava chování postav, vytvoření dalších úrovní, přidání dalších elementů do hry. S ohledem na časově dostupné možnosti této práce lze tyto návrhy uvažovat jako možnou budoucí práci na tomto projektu.

Kapitola 6

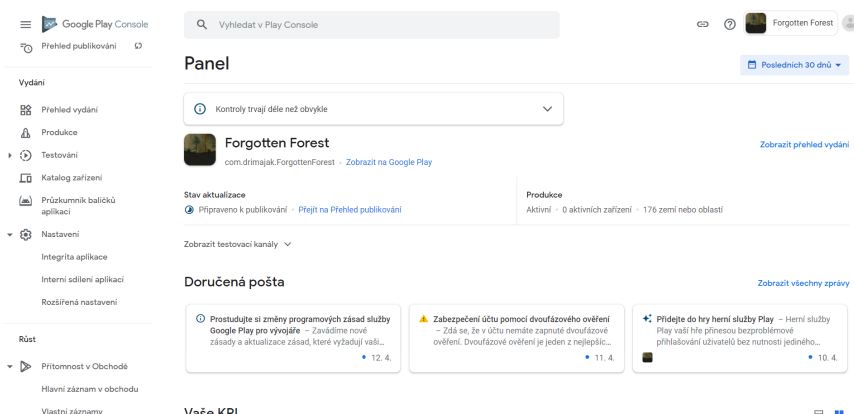
Vydání hry

6.1 Založení vývojářského účtu

Pro publikování hry v obchodě Google Play je potřeba si nejdříve založit vývojářský účet, pomocí kterého získáme přístup vývojářské konzole. Při zakládání účtu jsou po nás vyžadovány základní informace, jako je jméno, příjmení a telefonní číslo. Zároveň je nám při tomto procesu účtován jednorázový registrační poplatek ve výši 25\$. Po úhradě částky získáme přístup do konzole a můžeme začít vyplňovat informace o první aplikaci, kterou chceme v obchodě zveřejnit.

6.2 Vyplnění informací o hře

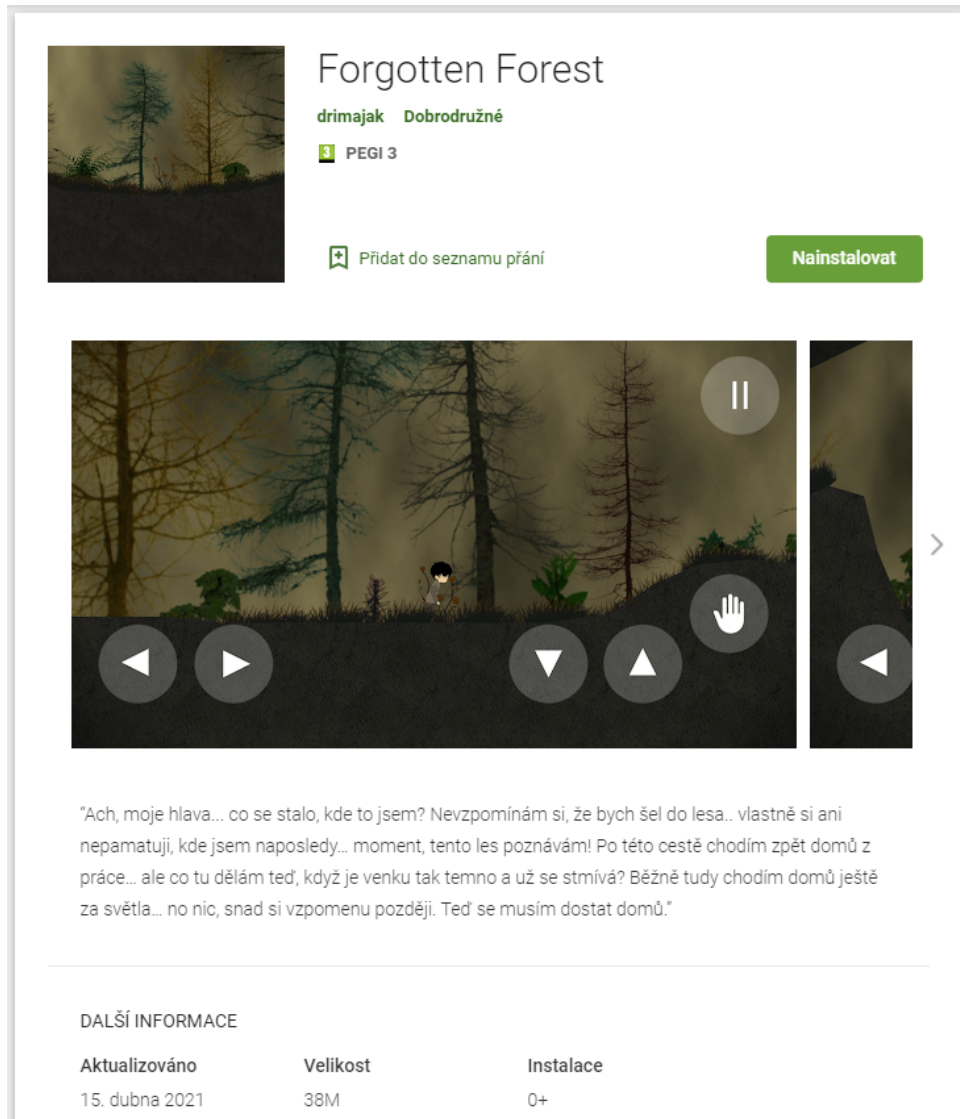
V konzoli se nám na počátku zobrazí spoustu nezbytných úkolů a dotazníků, které je třeba vyplnit, abychom hru mohli publikovat. Mezi tyto úkoly se řadí například vyplnění dotazníku věkové přístupnosti (content rating), kategorie aplikace, možnosti interakce s ostatními uživateli, zda hra obsahuje reklamy a nákupy, splňuje zásady a tak podobně. Dále v záložce *Záznam v obchodě* zadáváme propagační texty a grafické podklady hry, které budou přímo viditelné pro uživatele v obchodě Google Play.



Obrázek 6.1: Webové rozhraní Google Play Console

6.3 Vydání hry

Po vyplnění všech vyžadovaných údajů můžeme konečně začít s publikováním aplikace. Přes vývojářskou konzoli nahrajeme instalační soubor .apk. Pro dané vydání je možnost nastavit distribuci aplikace v obchodě pro jednotlivé části světa. Volitelně můžeme přidat i název verze či poznámky ke změnám v této verzi aplikace. Po ověření nahraného balíčku můžeme konečně zahájit vydání v produkčním kanálu. Od této chvíle je třeba čekat, než společnost Google bezpečnost aplikace prověří a také zkontroluje, zda dodržuje podmínky zásad pro vývojáře. Tento proces může trvat i několik dní. Po uplynutí této časové lhůty je aplikace veřejně publikována v obchodě Google Play a kdokoliv si ji může zobrazit a stáhnout.



Forgotten Forest
 drimajak Dobrodružné
 PEGI 3

Přidat do seznamu přání **Nainstalovat**

"Ach, moje hlava... co se stalo, kde to jsem? Nevzpomínám si, že bych šel do lesa.. vlastně si ani nepamatuji, kde jsem naposledy... moment, tento les poznávám! Po této cestě chodím zpět domů z práce... ale co tu dělám teď, když je venku tak temno a už se stmívá? Běžně tudy chodím domů ještě za světla... no nic, snad si vzpomenu později. Teď se musím dostat domů."

DALŠÍ INFORMACE

Aktualizováno	Velikost	Instalace
15. dubna 2021	38M	0+

Obrázek 6.2: Výsledná hra v obchodě Google Play



Kapitola 7

Závěr

Cílem této práce bylo navrhnout, implementovat a vydat mobilní hru na platformu Android ve vybraném herním engine. V první části byly porovnány nejnámější herní engine a následně byl proveden rozbor Unity engine. Následující kapitola popisuje celkový návrh hry a představuje Herní návrhový dokument popisující její hlavní rysy. Implementační část práce pojednává o způsobu realizace jednotlivých částí hry a přístupu řešení problematiky. Posléze je v další kapitole hra testována pomocí uživatelských testů. Na závěr jsou popsány náležitosti procesu a samotné vydání hry v obchodě Google Play.

Zadaný rozsah práce se podařilo úspěšně naplnit, výsledná hra je k dispozici na Google Play pro širokou veřejnost. Při zpracovávání práce nenastaly žádné větší komplikace, které by výrazně ovlivnily plánovaný rozsah práce.

Jako možné rozšíření do budoucna mě napadá několik věcí, které by mohly být do hry přidány. Jako první je vytvořit či dodat další grafické a zvukové podklady do hry. Při vývoji byly totiž využívány vlastní nebo volně dostupné prostředky a ne pro všechny herní elementy se podařilo požadovaný formát těchto podkladů získat. Jako další vylepšení by do hry mohl být implementován editor úrovní. Uživatelé by mohli vytvářet vlastní úrovně (například z předem předpřipravených bloků) a mohli je mezi sebou sdílet. Posledním návrhem je přidání hry více hráčů, kde by dva hráči mohli hrát společně na stejné mapě. Ta by byla uzpůsobena tak, že by si hráči museli navzájem pomáhat, aby se dostali skrz překážky.

Za pozitivní přínos práce hodnotím především nabytí nových znalostí z oblastí herního vývoje a užíváním herního engine, využití určitých matematických vědomostí v praxi, zjištění skutečné časové náročnosti projektu tohoto typu a osvojení si procesu distribuce a publikace mobilní aplikace pro veřejnost.



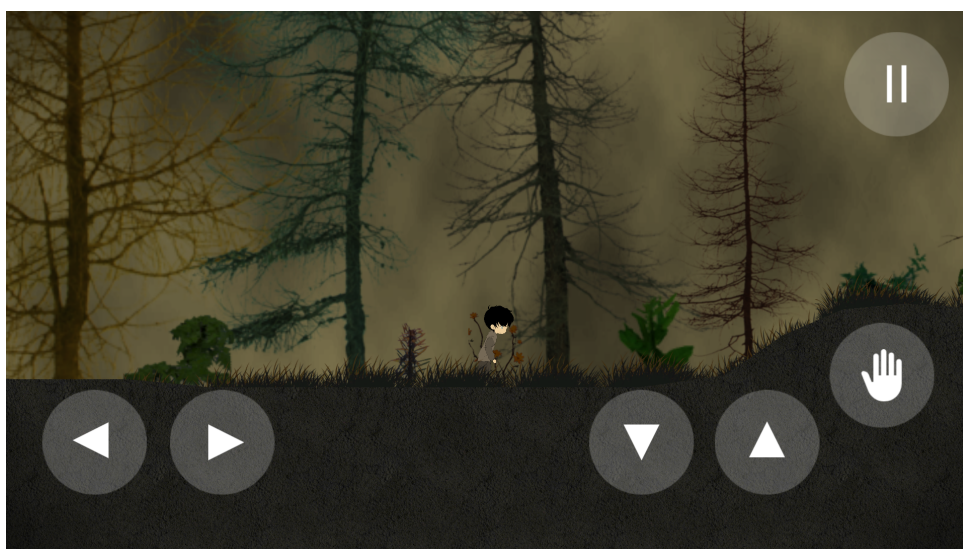
Literatura

- [1] Unity Technologies. Unity user manual, 2021. [Online. Accessed 2020-11-13]. Available at: <https://docs.unity3d.com/Manual/UsingTheEditor.html>.
- [2] Dali games. Lucid dream adventure 3 (google play), 2020. [Online. Accessed 2020-11-14]. Available at: <https://play.google.com/store/apps/details?id=games.dali.adventure.lucid.dream.part.three>.
- [3] uBeeJoy. Typoman mobile (google play), 2019. [Online. Accessed 2020-11-14]. Available at: <https://play.google.com/store/apps/details?id=com.ubj.typoman>.
- [4] Playdead. Limbo (google play), 2015. [Online. Accessed 2020-11-14]. Available at: <https://play.google.com/store/apps/details?id=com.playdead.limbo.full>.
- [5] Vladimir Beletsky. The mooseman (google play), 2018. [Online. Accessed 2020-11-14]. Available at: <https://play.google.com/store/apps/details?id=com.morteshka.mooseman>.
- [6] University of Melbourne. Mobile device accessibility, 2015. [Online. Accessed 2020-12-15]. Available at: <https://www.unimelb.edu.au/accessibility/presentations/mobile-device-accessibility>.
- [7] Anis Zarrad. Game engine solutions. *Simulation and Gaming*, 2018. Image edited. [Online. Accessed 2021-04-13]. Available at: https://www.researchgate.net/publication/323179176_Game_Engine_Solutions.
- [8] Ralph H Baer. *Videogames: in the beginning*. Rolenta Press, 2005.

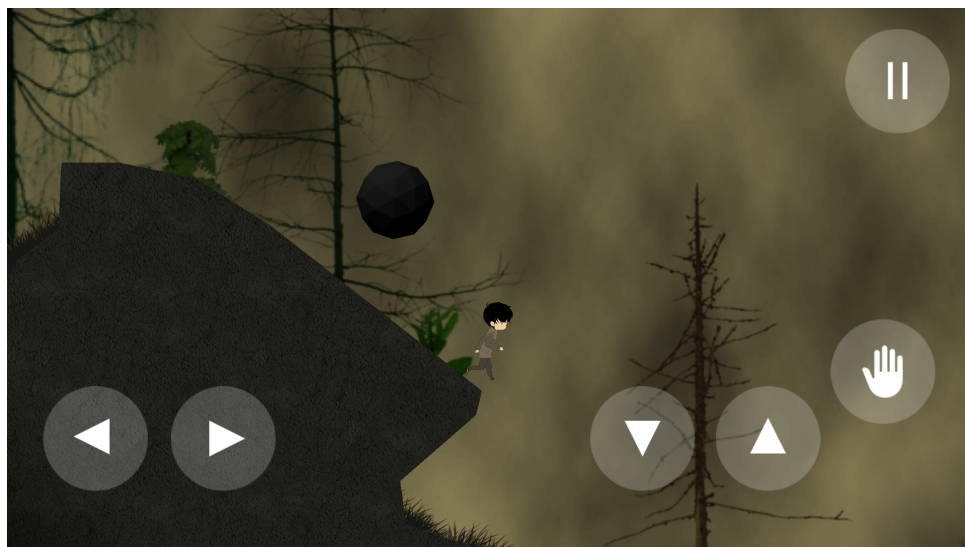
- [9] Rachel Kowert and Thorsten Quandt. *The Video Game Debate 2: Revisiting the Physical, Social, and Psychological Effects of Video Games*. Routledge, 2020.
- [10] Elina MI Koivisto. Mobile games 2010. In *ACM International Conference Proceeding Series*, volume 223. Citeseer, 2006. [Online]. Available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.69.7761&rep=rep1&type=pdf>.
- [11] Lauren Luz. 2020 mobile gaming industry overview, 2019. [Online. Accessed 2020-12-15]. Available at: <http://web.archive.org/web/20201202010942/https://www.ironsrc.com/blog/mobile-gaming-industry-trends-in-2020/>.
- [12] Collins Cobuild. Collins cobuild advanced learner’s english dictionary, 2008.
- [13] Unity Technologies. Unity plans comparison, 2021. [Online. Accessed 2020-11-14]. Available at: <https://store.unity.com/compare-plans>.
- [14] Harold Serrano. Game engine development, 2017. [Online. Accessed 2020-12-28]. Available at: <https://www.haroldserrano.com/blog/how-long-does-it-take-to-develop-a-game-engine>.
- [15] Mario Gonzalez Salazar, Hugo A Mitre, Cuauhtémoc Lemus Olalde, and José Luis González Sánchez. Proposal of game design document from software engineering requirements perspective. In *2012 17th International Conference on Computer Games (CGAMES)*. IEEE, 2012.
- [16] Jesse Schell. *The Art of Game Design: A book of lenses*. CRC press, 2008. [Online]. Available at: <http://www.sg4adults.eu/files/art-game-design.pdf>.
- [17] Terry Norton. *Learning C# by developing games with unity 3D*. Packt Publishing Ltd, 2013. [Online]. Available at: <http://www.computo-visual.org/Videojuegos/Libros/Learning%20C%23%20by%20Developing%20Games%20with%20Unity%203D.pdf>.

Příloha A

Obrázky ze hry



Obrázek A.1: Pochod lesem



Obrázek A.2: Padající balvan



Obrázek A.3: Pronásledující pavouk



Obrázek A.4: Bedna jako obrana přes šípem



Příloha B

Odkaz na hru

<https://play.google.com/store/apps/details?id=com.drimajak.ForgottenForest>



Příloha C

Projekt

Hra byla vytvořena v Unity engine verze 2019.4.19f1. Tuto verzi Unity lze stáhnout přes Unity Hub nebo na webových stránkách Unity. Pomocí Unity Hub lze pak následně přidat adresář projektu hry (Forgotten Forest) a ten následně otevřít v Unity. V případě, že Unity Hub nepoužíváme, můžeme projekt načíst přímo v Unity editoru (File -> Open project).

Projekt lze s velkou pravděpodobností otevřít i v novějších verzích Unity engine, nicméně hra byla vytvářena ve výše zmíněné verzi software a úplnou funkčnost lze tak zaručit pouze na ní.