

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství



**Česká
zemědělská
univerzita
v Praze**

Diplomová práce

Tvorba vzdělávací hry pro mobilní zařízení

Bc. Štěpán Marek

© 2024 ČZU v Praze

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Štěpán Marek

Informatika

Název práce

Tvorba vzdělávací hry pro mobilní zařízení

Název anglicky

Educational mobile game development

Cíle práce

Cílem teoretické části práce je seznámit čtenáře s mobilními hrami, popsat existující herní žánry a představit jaký mají přínos pro společnost. Dále je cílem teoretické části práce představit enginy pro tvorbu mobilních počítačových her a popsat základní postup vývoje 2D mobilních her.

Cílem praktické části je tvorba samotné 2D hry která bude určena pro mobilní zařízení. Pro vytvoření hry bude použit herní engine vhodný pro tvorbu mobilních her. Hra bude zobrazovat základní matematické operace díky kterým si uživatel bude zlepšovat matematické dovednosti. Za korektní splnění operace je hráč odměněn v podobě přičtení bodu do celkového skóre. Hra je vhodná pro žáky mateřských škol a prvního stupně základních škol.

Metodika

Prvním klíčovým krokem pro vytvoření diplomové práce bude nastudování odborných literárních zdrojů. V úvodu teoretické části práce se čtenář seznámí s mobilními hrami, jejich žánry a přínosem her pro společnost. Dále bude následovat popis enginů pro tvorbu mobilních her a samotný obecný postup vývoje mobilních her, který zahrnuje tvorbu konceptu hry, vývoj, implementaci a testování. V praktické části bude vytvořena mobilní hra a popsán postup vývoje dané hry. Hra bude zobrazovat základní početní operace, díky kterým bude uživatel rozvíjet své matematické dovednosti.

Doporučený rozsah práce

60–80 stran

Klíčová slova

mobilní hra, 2D, Unity, iOS, matematika

Doporučené zdroje informací

ALESSI, Patrick. Vývoj her pro iPhone a iPad: programování pro iOS. Brno: Zoner Press, 2012. ISBN 978-80-7413-199-8.

BUTTFIELD-ADDISON, Paris. Mobile Game Development with Unity. Spojené státy americké: O'Reilly Media, 2017. ISBN 9781491944745.

DORAN, John P. Unity 2022 Mobile Game Development. 3. vydání. Velká Británie: Packt Publishing, 2023. ISBN 180461372X.



Předběžný termín obhajoby

2023/24 LS – PEF

Vedoucí práce

Ing. Dana Vynikarová, Ph.D.

Garantující pracoviště

Katedra informačního inženýrství

Elektronicky schváleno dne 28. 11. 2023

Ing. Martin Pelikán, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 9. 2. 2024

doc. Ing. Tomáš Šubrt, Ph.D.

Děkan

V Praze dne 27. 03. 2024

Čestné prohlášení

Prohlašuji, že svou diplomovou práci „Tvorba vzdělávací hry pro mobilní zařízení“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 31.03.2024

Poděkování

Rád bych touto cestou poděkoval Ing. Daně Vynikarové, Ph.D., za vedení práce, poskytnutí cenných rad a za čas, který mi věnovala během tvorby této práce.

Tvorba vzdělávací hry pro mobilní zařízení

Abstrakt

Diplomová práce se zabývá tvorbou vzdělávací hry pro mobilní zařízení. V teoretické části je uveden popis mobilních her a jejich historie. Tato část dále popisuje monetizaci mobilních her, včetně různých modelů jako jsou například reklamy. Také jsou zde představeny žánry a kategorie mobilních her. Dále je práce zaměřena na popis přínosů, které mobilní hry přinášejí pro společnost, včetně sociální interakce a vzdělávání. Další část popisuje vhodné enginey pro vývoj mobilních her a proces vývoje hry včetně plánování, analýzy požadavků, tvorby wireframů a vývojových diagramů. Praktická část se soustředí na vývoj vzdělávací mobilní hry. Tato část obsahuje detailní popis procesu vývoje hry v engineu Unity, včetně popisu kódu. Zde je zahrnutý také plán hry, funkční a nefunkční požadavky, Use-case diagram, wireframy a vývojový diagram. Na závěr práce je celá hra otestována na platformách iOS a Android. Výsledná hra bude zobrazovat jednoduché matematické příklady, které bude uživatel počítat. Výsledky bude určovat pomocí tlačítek s výsledky. Jeden výsledek je vždy správný. Pokud uživatel zvolí správně, hra mu přičte bod do skóre. Pokud zvolí špatně, ztratí pokus na vyřešení příkladu.

Klíčová slova: mobilní hra, 2D, Unity, iOS, matematika

Development of an educational game for mobile devices

Abstract

The diploma thesis deals with the creation of an educational game for mobile devices. The theoretical part contains a description of mobile games and their history. This section further describes the monetization of mobile games, including different models such as ads for example. Genres and categories of mobile games are presented as well. Furthermore, the work focuses on the benefits that mobile games bring to society, including social interaction and education. The next section describes suitable mobile game development engines and the game development process including planning, requirements analysis, creation of wireframes and flowcharts. The practical part focuses on the development of an educational mobile game. This section contains a detailed description of the game development process in the Unity engine, including a description of the code. There is also included the game plan, functional and non-functional requirements, Use-case diagram, wireframes, and flowchart. At the end of the work, the entire game is tested on the iOS and Android platforms. The resulting game will display simple math examples for the user to calculate. The results will be determined using the results buttons. One result is always correct. If the user chooses correctly, the game adds a point to the score. If they choose wrongly, they lose their attempt to solve the example.

Keywords: mobile game, 2D, Unity, iOS, math

Obsah

1 Úvod	12
2 Cíl práce a metodika	13
2.1 Cíl.....	13
2.2 Metodika	13
3 Teoretická východiska	14
3.1 Mobilní hry	14
3.1.1 Historie	14
3.1.2 Monetizace v mobilních aplikacích	15
3.1.2.1 Placené hry	15
3.1.2.2 Hra zdarma s reklamou.....	15
3.1.2.3 Hra s mikrotransakcemi (Freemium).....	16
3.1.3 Kategorie mobilních her	16
3.1.3.1 Hyper-casual hry	17
3.1.3.2 Hybrid-casual hry	17
3.1.3.3 Casual hry	17
3.1.3.4 Mid-core a Hard-core hry	18
3.1.4 Žánry mobilních her.....	18
3.1.5 Přínos her pro společnost.....	19
3.1.5.1 Posilování kognitivních dovedností	19
3.1.5.2 Lepší motorika.....	19
3.1.5.3 Zvýšená koncentrace	20
3.1.5.4 Uvolnění stresu.....	20
3.1.5.5 Vzdělávání pomocí her.....	20
3.1.6 Výukové hry	20
3.1.6.1 Matemág.....	21
3.1.6.2 Lightbot : Code Hour	21
3.2 Herní enginey.....	22
3.2.1 Unity	23
3.2.2 Unreal Engine	24
3.2.3 Cryengine.....	24
3.2.4 Solar2D (Corona SDK).....	25
3.2.5 SpriteKit.....	25
3.2.6 Cocos2D	26

3.2.7	BuildBox.....	26
3.2.8	Construct 3.....	26
3.2.9	GameMaker Studio 2.....	27
3.2.10	MonoGame.....	27
3.2.11	Haxe.....	27
3.2.12	Gideros.....	28
3.2.13	Godot.....	28
3.2.14	Další herní enginy.....	28
3.3	Proces vývoje mobilní hry.....	29
3.3.1	Nápad a plán na vývoj hry.....	29
3.3.1.1	Atributy dobré hry.....	29
3.3.2	Vhodná platforma a engine.....	30
3.3.3	Rozdíly mezi vývojem her pro mobilní zařízení a počítače.....	31
3.3.4	Game design dokument.....	32
3.3.5	Monetizace.....	33
3.3.6	Design hry a wireframů.....	33
3.3.6.1	Use-case diagram.....	33
3.3.6.2	Vývojový diagram.....	35
3.3.7	Vývoj hry.....	37
3.3.7.1	Programovací jazyk.....	37
3.3.7.2	Jazyk C#.....	37
3.3.8	Nahrání hry do obchodu a tvorba aktualizací.....	39
4	Tvorba mobilní hry.....	40
4.1	Plán hry.....	40
4.2	Funkční a nefunkční požadavky.....	40
4.3	Use-case diagram.....	41
4.4	Wireframy.....	42
4.5	Vývojový diagram.....	45
4.6	Vhodný engine.....	46
4.7	Vývoj hry.....	48
4.7.1	Příprava scén.....	48
4.7.2	Skripty.....	51
4.8	Grafika hry.....	61
4.9	Testování na mobilním zařízení.....	63
5	Výsledky a diskuse.....	66
6	Závěr.....	67
7	Seznam použitých zdrojů.....	68

8 Přílohy	71
8.1 Ikonka hry	71
8.2 Scény hry.....	71

Seznam obrázků

Obrázek 1: Historie mobilních her (2).....	15
Obrázek 2: Hra Matemág (17).....	21
Obrázek 3: Hra Lightbot (18)	22
Obrázek 4: Příklad use-diagramu (31).....	34
Obrázek 5: Terminál (34)	36
Obrázek 6: Krok algoritmu (34)	36
Obrázek 7: Výpis zprávy (34).....	36
Obrázek 8: Podmínka (34).....	37
Obrázek 9: Use-case diagram mobilní hry	42
Obrázek 10: Wireframe pro úvodní obrazovku hry.....	43
Obrázek 11: Wireframe pro nastavení hry.....	43
Obrázek 12: Wireframe samotné hry.....	44
Obrázek 13: Wireframe pro konec hry	45
Obrázek 14: Vývojový diagram.....	46
Obrázek 15: Unity hub.....	47
Obrázek 16: Instalace modulů	47
Obrázek 17: Šablony pro projekt.....	48
Obrázek 18: Rozhraní unity	48
Obrázek 19: Nastavení buildu.....	49
Obrázek 20: První scéna	50
Obrázek 21: Druhá scéna.....	50
Obrázek 22: Hlavní scéna hry.....	51
Obrázek 23: Přiřazení skriptu pro tlačítko start hry	52
Obrázek 24: Přiřazení objektů k proměnným	54
Obrázek 25: Přiřazení proměnných	59
Obrázek 26: Objekty obrazovky pro konec hry.....	60
Obrázek 27: Styl písma.....	61
Obrázek 28: Příklad zobrazení stylu písma	62
Obrázek 29: Design tlačítek.....	62
Obrázek 30: Pozadí hry.....	63
Obrázek 31: Testování na zařízení iOS	64

Seznam tabulek

Tabulka 1: Funkční a nefunkční požadavky	41
Tabulka 2: Zařízení určené k testování.....	63

Seznam kódů

Kód 1: Skript pro úvodní obrazovku	52
Kód 2: Ukládání nastavení.....	53
Kód 3: Definice proměnných.....	53
Kód 4: Funkce pro generování matematického příkladu.....	55
Kód 5: Genrování příkladů	55
Kód 6: Pokračování funce pro generování příkladů	56
Kód 7: Funkce pro tlačítka s výsledky.....	56

Kód 8: Funkce pro smazání příkladu, ztrátu životů a zvýšení skóre	57
Kód 9: Skript pro řízení časomíry.....	59
Kód 10: Definování proměnných pro časomíru	60
Kód 11: Konec hry a funkce pro tlačítka.....	60
Kód 12: Start hry a kontrola životů	61

Seznam příloh

Příloha 1: Ikonka.....	71
Příloha 2: Úvodní menu.....	71
Příloha 3: Nastavení.....	72
Příloha 4: Počítání příkladů	72
Příloha 5: Konec hry	73

1 Úvod

V dnešní době jsou mobilní hry neodmyslitelnou součástí digitálního světa, který neustále ovlivňuje naši společnost a způsob, jakým trávíme svůj volný čas. Mobilní hry se staly nejen zábavnou formou relaxace, ale i prostředkem pro vzdělávání a rozvoj. Tato diplomová práce se zaměřuje na mobilní hry, popis herních enginů a na proces tvorby mobilních her. Součástí práce je také tvorba vzdělávací matematické mobilní hry, která je předmětem praktické části.

Teoretická část této práce začíná přehledem toho, co vlastně mobilní hry jsou a jakým historickým vývojem prošly od vzniku první mobilní hry. Dále se práce zabývá důležitým aspektem monetizace mobilních her. V rámci této části budou rozebrány různé modely monetizace, jako jsou reklamy nebo in-game nákupy. V další části budou popsány kategorie a žánry mobilních her. Zároveň je práce zaměřena na přínosy, které mobilní hry přináší pro společnost, včetně rozvoje kognitivních dovedností, sociální interakce a vzdělávání. Další část práce se věnuje procesu vývoje mobilních her. Popisuje vhodné enginy pro vývoj, proces vývoje hry včetně plánování, analýzy požadavků, tvorby wireframů a vývojových diagramů.

Praktická část práce se soustředí na konkrétní vývoj vzdělávací matematické mobilní hry. Obsahuje plán hry, funkční a nefunkční požadavky, Use-case diagram, wireframy pro hru a vývojový diagram. Na základě toho je vyvíjena mobilní hra v herním enginu Unity. V této části práce je také popsán proces vývoje hry, včetně detailního popisu kódu. V závěru práce bude provedeno testování matematické hry na mobilních platformách iOS a Android, které ověří funkčnost aplikace.

2 Cíl práce a metodika

2.1 Cíl

Cílem teoretické části práce je seznámit čtenáře s mobilními hrami, popsat existující herní žánry a představit jaký mají přínos pro společnost. Dále je cílem teoretické části práce představit enginy pro tvorbu mobilních počítačových her a popsat základní postup vývoje 2D mobilních her. Cílem praktické části je tvorba samotné 2D hry která bude určena pro mobilní zařízení. Pro vytvoření hry bude použit herní engine vhodný pro tvorbu mobilních her. Hra bude zobrazovat základní matematické operace díky kterým si uživatel bude zlepšovat matematické dovednosti. Za korektní splnění operace je hráč odměněn v podobě přičtení bodu do celkového skóre. Hra je vhodná pro žáky mateřských škol a prvního stupně základních škol.

2.2 Metodika

Prvním klíčovým krokem pro vytvoření diplomové práce bude nastudování odborných literárních zdrojů. V úvodu teoretické části práce se čtenář seznámí s mobilními hrami, jejich žánry a přínosem her pro společnost. Dále bude následovat popis enginů pro tvorbu mobilních her a samotný obecný postup vývoje mobilních her, který zahrnuje tvorbu konceptu hry, vývoj, implementaci a testování. V praktické části bude vytvořena mobilní hra a popsán postup vývoje dané hry. Hra bude zobrazovat základní početní operace, díky kterým bude uživatel rozvíjet své matematické dovednosti.

3 Teoretická východiska

3.1 Mobilní hry

Hry pro mobilní zařízení a tablety představují specifickou kategorii interaktivních digitálních her, které jsou vyvinuty s cílem poskytnout zábavu na chytrých telefonech a tabletech. (1)

Mobilní hry si vybudovaly velikou popularitu díky snadnému přístupu, protože v dnešní době skoro každý vlastní chytrý telefon. Dají se snadno stáhnout v obchodě s aplikacemi, který je v každém moderním telefonu. V něm je nespočet her zadarmo nebo za nízkou cenu. Mobilní hry jsou oblíbené také díky sociálnímu prvku. Mnoho z nich je možné hrát s přáteli nebo s dalšími hráči. To dává hráčům možnost různě spolu interagovat, soutěžit proti sobě anebo spojovat síly. (1)

3.1.1 Historie

První mobilní hra, Tetris, spatřila světlo světa v roce 1994 na telefonu Hagenuk MT-2000, který se stal prvním zařízením s hrou i zabudovanou anténou. O tři roky později přišla hra Had pro Nokii 6110, popularita mobilních her však výrazně vzrostla až s telefonem Nokia 3310, který hru Had vylepšil o lepší grafiku a možnost ukládání skóre.

Další úspěšná hra, jako Alien Fish Exchange přišla s rozvojem technologie WAP, umožňující připojení k internetu. V roce 2001 se objevila hra Space Invaders s arkádovými prvky. S nástupem zařízení, které umožňovalo lepší zobrazení barev se do mobilního herního průmyslu zapojily i velké společnosti, jako EA, která v roce 2004 vydala závodní hru Asphalt GT. (2)

Revoluci přinesl iPhone v roce 2007, s nímž se objevil App Store s prvními 500 aplikacemi včetně her jako Texas Hold'em a Super Monkey Ball. Dotyková obrazovka otevřela cestu novým herním směrům a v roce 2009 přišla hra Angry Birds, která díky své jednoduchosti a zábavě rychle získala popularitu. Následně byla uvedena i na Android s verzí s reklamami, které bylo možné zaplaceným poplatku odstranit. (2), (3)

V roce 2012 se objevila hra Candy Crush, která se stala jednou z nejdělečnějších her, a to hlavně díky příjmům z reklam a in-game nákupům. (4)

V roce 2016 byla vydána hra Pokémon Go, která v době vydání zažila obrovský úspěch, v počtu uživatelů překonala několik sociálních sítí jako Twitter, Tinder nebo Snapchat. Od té doby vzniklo mnoho dalších populárních her, některé z nich byly původně

určeny pro PC a později byly adaptovány pro mobilní zařízení, jako například Fortnite, Genshin Impact nebo Among Us. (2), (5)



Obrázek 1: Historie mobilních her (2)

3.1.2 Monetizace v mobilních aplikacích

„Monetizace mobilních her je proces, při kterém získávají vydavatelé peníze z mobilních her.“ (6)

Existují tři populární způsoby jak hru monetizovat. Jedním ze způsobů, jak monetizovat hru je vydat ji přímo placenou, za její stažení je tedy nutné zaplatit. Dalším populárním způsobem je nabízet hru v obchodě zadarmo a mít v ní zabudovanou reklamu. Poslední je Freemium model, který hru nabízí zadarmo a dává uživateli možnost zaplatit si ve hře jisté výhody, třeba rychlejší postup ve hře nebo herní měnu. (6)

3.1.2.1 Placené hry

Jedním z modelů jak hru nebo aplikaci monetizovat je zpoplatněné stažení. Model placených her se využívá velice málo a považuje se za riskantní. Tyto hry totiž generují zisk spíše z počátku vydání aplikace a zároveň nejsou tolik populární jako hry na konzole nebo počítače. Tento model se vyplatí spíše u již populárních her, které existují třeba právě na počítači. Pokud by se ale i tak přistoupilo k této strategii monetizace, je doporučeno vydat hru ještě navíc s lite verzí hry, na které si uživatelé vyzkoušejí, zda se jim hra líbí nebo ne. (6)

3.1.2.2 Hra zdarma s reklamou

Další velmi často využívanou strategií, je vydat hru kompletně zadarmo ale s nějakou formou reklamy. Na reklamy ve hrách totiž lidé klikají častěji než na reklamy na webu,

zvlášť když za to uživatel získá nějakou odměnu, třeba herní měnu nebo možnost dalšího postupu ve hře. Reklama ve hře ale musí být ve vyvážené míře, pokud by uživatele příliš zahltila, mohl by hru vypnout nebo úplně odstranit ze svého zařízení. Způsobů, kterými je možné reklamu zobrazovat je několik tohle jsou některé z nich:

- **„Přímé reklamy:** *Reklama je založená na dohodě se sponzory, nebo ostatními aplikacemi a vzájemné propagaci.*
- **Bannerové reklamy:** *Reklama je umístěná v horní nebo dolní části displaye a zobrazuje drobné obrázkové nebo textové reklamy.*
- **Mediální reklamy:** *Reklama má formu videa se zvukem a využívá další mediální elementy pro získání pozornosti hráče. Většinou je v rohu odpočítáván čas, po který musí reklama běžet.*
- **Odměňující reklamy:** *Shlédnutí reklamy (většinou videa) odmění hráče herní měnou, předměty, upgrady atd.*
- **Reklamy za základě události:** *Reklama se spustí ve chvíli, kdy hráč dorazí do určitého bodu hry (místa, levelu, nahraného skóre). Součástí tohoto druhu reklamy je většinou i odměna pro hráče.“ (6)*

3.1.2.3 Hra s mikrotransakcemi (Freemium)

Tento model monetizace je využíván nejčastěji, uživatel může hru stáhnout zadarmo a přímo ve hře má možnost zaplatit nějaký druh výhody. Výhod, které si hráč může koupit je několik většinou v podobě loot boxů, které vypadají jako krabice, truhly anebo balíčky s kartami. Obsah těchto balíčků je náhodně generován, a tak není předem jisté co hráč dostane. Další výhody přicházejí v podobě měsíčních nebo týdenních poplatků a hráči umožní další herní čas nebo hraní bez reklam. Platby za herní čas se u mobilních her neobjevují tak často, spíše na ně můžeme narazit u větších titulů pro počítač, jako tomu je třeba u hry World of Warcraft. (6)

3.1.3 Kategorie mobilních her

Mobilní hry nabízejí velikou řadu herních žánrů a každý nabízí uživateli odlišný požitek ze hry. Mobilní hry by se také daly rozdělit do pěti kategorií, jimiž jsou Hard-core, Mid-core, Casual, Hybrid-casual, Hyper-casual hry. (7)

3.1.3.1 Hyper-casual hry

Hyper-casual hry jsou vytvářeny tak, aby byly co nejjednodušší a rychlé v průběhu hraní. Hry v této kategorii jsou většinou postavené na jednoduché mechanice, která ke hraní vyžaduje jenom občasné klepnutí na obrazovku zařízení. Jedná se v podstatě o takzvané „tap-to-play“ hry. Tyto jednoduché hry nepřinášejí žádný tutoriál pro hráče, jsou vhodné na krátké přestávky a pro lidi, které se nedokážou soustředit po delší časový úsek.

Jsou designované tak aby, během krátké doby nutily uživatele vracet se zpět pro více obsahu. Díky tomu je u těchto her velký počet možností jak hru monetizovat. U některých her byl tento typ kategorie dohnán až do extrému, v některých aplikacích je možnost přestat klikat na obrazovku úplně, ale i tak bude hra pokračovat dál bez nutnosti interakce od uživatele. Do téhle kategorie je možné zařadit například hru Cats & Soup. (7), (8)

3.1.3.2 Hybrid-casual hry

Další kategorie mobilních her nabízí kombinaci dvou různých žánrů. Cílem této kategorie je zachovat si jednoduchost hry jako je tomu u Hyper-casual her, ale zároveň nabídnout uživateli více možností ke koupi dalšího obsahu nebo výhod ve hře. V podstatě tedy zachovává stejné „tap-to-play“ hraní jako tomu u hyper-casual her, ale zároveň nabízí uživateli o trochu složitější herní mechaniky a funkce.

Hry této kategorie nabízí uživatelům systém pro sbírání určitého obsahu ve hře a složitou herní ekonomiku. Díky těmto prvkům je možné, aby si uživatel vybudoval nějaký cíl, což způsobuje jeho konstantní návrat ke hře.

Do této kategorie spadá třeba hra od vývojářů z Lion Studios s názvem Ancient Battle. Je to jednoduchá bitevní simulace, ve které hráči sbírají zlato, kouzla, drahokamy a karty a objevují nové země s cílem umístit se co nejvýše v celosvětovém žebříčku. (8)

3.1.3.3 Casual hry

Tento typ kategorie se považuje za ten nejoblíbenější v herním průmyslu, na zařízení Android tento typ her hraje přibližně 58 % uživatelů. Hry tohoto typu se snadno hrají, nejsou složité na pochopení a nevyžadují dlouhodobé hraní, nejsou však tak jednoduché jako Hyper-casual hry. Jejich zisk je generován především z reklam a nákupů obsahu v aplikaci. Do této kategorie můžeme zařadit hry jako Township, CandyCrush, Temple Run, Fruit Ninja, Subway Surfers, Crossy Roads. (7)

3.1.3.4 Mid-core a Hard-core hry

Mid-core kategorie spadá někdy mezi Casual a Hardcore-hry. Obě kategorie vyžadují více času stráveného ve hře a často nabízejí hraní více hráčů. Do obou kategorií spadají spíše strategické hry, RPG, Moba hry a mezi které je možné zařadit hry jako Clash of Clans nebo Brawl Stars. (9)

3.1.4 Žánry mobilních her

- **Akční hry** – Tento žánr je populární spíše na počítače a konzole, protože dynamické prvky těchto her nejsou příliš vhodné pro malé obrazovky. Akční hry jsou často spojovány s dalšími žánry, mezi které patří: Střílečky, Plošinovky, Stealth, Battle Royale. (10)
- **Dobrodružné** – Za dobrodružné hry lze považovat jednoduše ty, ve kterých se hráč vydává na nějaké dobrodružství, které je zároveň často spojováno s řešením určitých hádanek. (11)
- **Arkády** – Tento žánr je považován za velice návykový. Sem spadají hry, které disponují jednoduchým ovládním, hratelností a často fungují v nekonečné smyčce. Hráč se snaží dosáhnout nejvyššího skóre, a pokud selže, musí začít znovu. (11)
- **Battle Royale** – Tento typ her zažil v posledních letech velký úspěch. Jedná se o hry, ve kterých hráč (sám nebo v týmu) bojuje proti ostatním hráčům v nějakém virtuálním světě a ten kdo zůstane jako poslední, vyhrává. (11)
- **Dětské** – Tento typ žánru se také nazývá edukativní. Tento typ her je určený pro malé děti, s cílem něčemu je naučit. (11)
- **Multiplayer online battle arena (MOBA)** – Jedná se o polární žánr, zejména díky hře League of Legends. Jde o hry, ve kterých tým hráčů bojuje proti jinému týmu. Ve většině těchto her se tým skládá z pěti členů, každý hraje za předem vybraného hrdinu a společně se snaží zničit hlavní základnu druhého týmu. (10)
- **Role-playing (RPG)** – RPG (Role playing game) nebo MMORPG (Massively multiplayer online role-playing games) jsou hry, ve kterých hráči hrají za postavy, co si sami vytvářejí. S nimi potom prozkoumávají virtuální svět, interagují s ostatními hráči a vylepšují svoji postavu. (10)
- **Závodní** – Žánr her, který často nabízí velký výběr motorek nebo aut. V těchto hrách hráči závodí proti sobě nebo proti oponentům vytvořených od samotné hry. (10)

- **Simulace** – Žánr, který simuluje reálné životní situace. (10)
- **Strategie** – V těchto hrách se hráči rozhodují, jaký bude jejich další krok, často je tento žánr spojován s vojáky a bitevními poli. (10)
- **Sport** – Do tohoto žánru spadají hry, které nabízejí sportovní téma. Známostou hrou tohoto žánru je Fifa. (11)
- **Další** – Puzzle, Karetní, Casino. (11)

3.1.5 Přínos her pro společnost

Hraní mobilních her má pozitivní dopad na duševní zdraví člověka. Hraním mobilních her lze nejenom vydělávat peníze, ale také je díky nim možné rozvíjet kreativní myšlení. Dokážou také léčit lidské tělo a zvládat duševní choroby. Zde je pár benefitů, které plynou z hraní mobilních her a her obecně. (12)

3.1.5.1 Posilování kognitivních dovedností

Hlavním přínosem hraní mobilních her je jejich schopnost zlepšovat kognitivní dovednosti. Jedná se o efektivní a zábavnou metodu pro trénování mozku. Hráči jsou vyzváni, aby hledali způsoby nebo strategie, jak vyhrát a neustále řešili problémy či překážky, které jim hra nastavuje. Díky tomuto lze zlepšovat kognitivní funkce, protože je vyžadováno logické myšlení při každém rozhodování, které hráč učiní. (12)

Dle studie z roku 2013 dokážou videohry zlepšit dovednost pro řešení problému. To platí u her s otevřeným světem, ve kterém hráči plní úkoly a řeší tak hádanky nebo problémy, které jsou s nimi spojené.

Z výzkumu, který proběhl v roce 2022, bylo zjištěno, že děti, které strávily hraním her zhruba 21 hodin týdně, vykazovaly lepší kognitivní schopnosti, zejména ve funkcích spojených s pamětí, ve srovnání s dětmi, které nehrály vůbec. (13)

3.1.5.2 Lepší motorika

Časté hraní mobilních her pomáhá také při rozvoji motoriky. Na rozdíl od uživatelů, kteří se hraní mobilních her nijak nevěnují, vykazují pravidelní hráči lepší reakční dobu a vylepšenou motoriku. (12)

Hraní má pozitivní vliv na šedou hmotu mozku, což je klíčová část mozku odpovědná za motoriku, paměť a emocionální reakce. Studie z roku 2015, která srovnávala expertní

hráče akčních videoher s nováčky, ukázala, že ti, kteří mají s hraním her bohaté zkušenosti, vykazovali vyšší množství šedé hmoty a lepší propojení mozkových funkcí. (13)

3.1.5.3 Zvýšená koncentrace

Soustředění na určité úkoly je důležité, a to ať už při studiu nebo při práci. K tomu mohou přispět právě mobilní hry, které zlepšují schopnost se koncentrovat. Od hráčů je vyžadováno, aby se na hraní soustředili delší dobu, což jim pomáhá udržovat pozornost. Mobilní hry se také dají využít pro pomoc dětem, které trpí ADHD nebo autismem. Dále se pomocí nich dají zlepšovat i sociální dovednosti. (12)

3.1.5.4 Uvolnění stresu

Hry jsou také dobrým nástrojem pro zlepšení duševní pohody. Stres vede k depresi, ale zábavné činnosti jako je hraní her mohou s těmito problémy efektivně pomoci. Mobilní hry také umožňují introvertům snadněji navazovat kontakt s lidmi pomocí hraní s ostatními hráči ve virtuálním světě. Hry tak dokážou spojovat miliardu lidí po celém světě s cílem dosažení určitého úspěchu ve hře. (12)

3.1.5.5 Vzdělávání pomocí her

Hry mají několik benefitů a jednou z nich je také pomoc při vzdělávání dětí. Spousta z nich je totiž vzdělávacího charakteru, s cílem uživatele něco naučit. Tyto hry disponují skvělou zábavou a často i poutavou grafikou, což pomáhá dětem se učit. Pokud je do učení zapojena grafika nebo obrázky, často je snazší si danou věc zapamatovat. (12)

3.1.6 Výukové hry

Výukové hry patří do žánru her, který se skládá z prvků vhodných pro vzdělávání uživatele. Tento typ her kombinuje učební prvky s hraním, což umožňuje hráčům učit se zábavným způsobem. Tento přístup vytváří atraktivní a efektivní prostředek pro učení a výuku.

Vzdělávací hry spojují prvky zábavy s učebním obsahem, což hráčům poskytuje zábavnější a lepší zážitek z učení. To vede ke snazšímu a pevnějšímu vstřebání informací. Tyto hry jsou specificky vytvořené s cílem poskytovat určitou formu vzdělávání, díky čemuž jsou považovány za efektivní nástroje pro rozvoj vědomostí. (14)

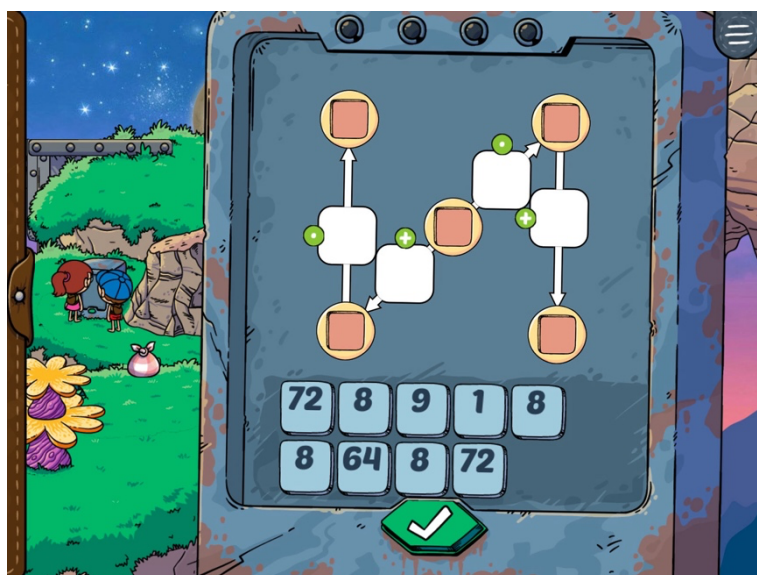
Tento vzdělávací proces je založený na třech faktorech:

- **Oživuje to vzdělávání** – Mění učení do zábavné a vzrušující hry bez nudných lekcí. Díky tomu si studenti snadno zapamatují informace, aniž by si to sami uvědomovali.
- **Vzbuzují větší zájem** – Uživatelé jsou během hraní odměňováni medailemi nebo jinými bonusy. To přitáhne jejich pozornost a zvětší zájem o učení.
- **Procvičování** – Uživatelé získávají cenné zkušenosti, bez jakéhokoliv nebezpečí. To se týká například leteckých simulátorů. (15)

Vzdělávací hry mohou mít několik forem, mezi které patří například simulace, kvízy, logické hry nebo jazykové aplikace. Následující aplikace byly vybrány, protože vhodně reprezentují edukativní hry. (16)

3.1.6.1 Matemág

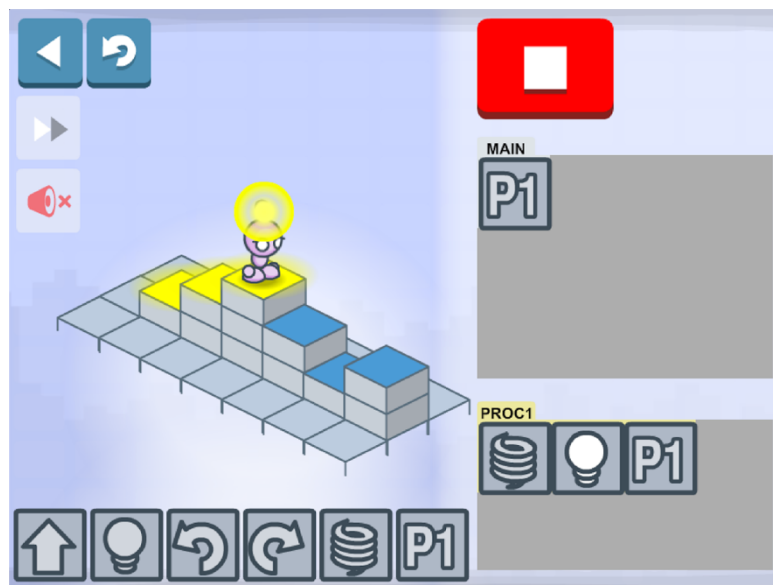
Hra Matemág je atraktivní matematická hra od českých tvůrců inspirovaná Hejného metodou. Hra je plně v češtině a obsahuje mluvený příběh, který zavede děti do světa plného matematických kouzel. V tomto světě se děti setkávají s logickými hádankami a matematické úlohy. Hra je vhodná pro děti od pěti let. (16)



Obrázek 2: Hra Matemág (17)

3.1.6.2 Lightbot : Code Hour

Tato hra podporuje logické a analytické myšlení. Uživatel využívá jednoduché příklady k řízení robota po různých trasách tak aby zapnul modrá pole. Je to skvělý prostředek, jak se děti mohou seznámit s principy programování a zároveň si procvičit prostorovou orientaci. Hra je vhodná pro děti od šesti let. (16)



Obrázek 3: Hra Lightbot (18)

3.2 Herní engine

„Jako engine se v herním vývoji označuje program, ve kterém se vytváří daná hra. Jde o komplexní a víceméně jednotný soubor nástrojů pro vývojáře, pomocí kterých hra získává jednotný tvar a může fungovat jako celek.“ (19)

Herních engineů je několik, přičemž každý má jiná pravidla, a i jinak funguje. Herní enginey uživatelům usnadňují práci při tvorbě samotné hry. V podstatě jde o jakousi předpřipravenou formu hry, na které vývojář dále vytváří vlastní hru. Enginey dokážou vykreslovat grafiku, provádět fyzikální simulace, manipulovat se světlem a se zvukem. Usnadňují také převod hry na jiné herní platformy a často mají předpřipravenou sadu prvků (např. 3D modely, skeny terénu), které může vývojář využívat.

Enginey pracují s programovacími jazyky, často se C# nebo C++, vždy, ale záleží na tom, které jazyky daný engine podporuje.

Každý herní engine je trochu jiný, kvůli tomu se stává, že nevyhovuje potřebám studia. Proto si mnoho herních studií vytváří vlastní, v tomto případě se jedná o proprietární enginey. Mezi ně patří například RAGE, Anvil nebo Creation Engine. V některých případech si studia enginey půjčují, a to za určitý poplatek. Další možností jsou komerční nástroje, ty jsou dostupné buďto zadarmo anebo za nějaký poplatek. Mezi nejznámější patří Unreal Engine od firmy Epic Games. Ten využívají jak velká studia, tak i nezávislí vývojáři, protože nabízí velkou řadu funkcí, kvalitní grafiku, snadnou manipulaci i optimalizaci a ke všemu ještě licenci zadarmo. Mezi ty snadnější enginey lze zařadit Unity, který je taktéž k dispozici

zdarma. Je vhodný spíše pro menší vývojáře, nenabízí tolik funkcí a nemá tak propracovanou grafiku.

Mezi optimální enginy pro střílečky a hry z pohledu první osoby se často používá CryEngine od společnosti CryTek. Disponuje vhodnými nástroji pro tvorbu hry z první osoby a nabízí velký grafický potenciál. Hry vyvíjené v tomto enginu, ale často vyžadují velký výpočetní výkon.

Existují také amatérské enginy, které umožňují tvorbu hry bez větších programátorských dovedností. Mezi ně patří například Game Maker. (19)

Herní enginy uvedené níže, ať už jsou placené či dostupné zdarma, jsou hodnoceny jako nejlepší vývojové platformy pro rok 2023 a 2024. Taktéž jsou vysoce doporučovány a patří mezi vhodné nástroje pro začátečníky, ale zároveň i pro zkušené odborníky. (22)

3.2.1 Unity

Jeden z nejpobulárnějších herních enginů je Unity, který používá přibližně 47 % herních vývojářů. Byl vytvořený společností Unity Technologies a je známý především díky skvělým funkcím a cenové dostupnosti. Je to skvělý nástroj, protože umožňuje multiplatformní vývoj her, je tedy možné v něm vytvářet hry pro počítače, mobily, konzole, a i pro zařízení, která využívají rozšířenou a virtuální realitu. Unity je vhodným nástrojem pro vývoj nezávislých her a také pro začínající vývojáře. Umožňuje vyvíjet 2D a 3D hry. Nabízí řadu užitečných funkcí, mezi které patří vykreslování v reálném čase, fyzikální simulace a nabízí také nástroje pro animaci. Dále nabízí rozsáhlý obchod, ve kterém lze najít již vytvořené modely nebo pluginy, vhodné pro vylepšení her. Unity podporuje dva programovací jazyky. Jedním je C# a tím druhým je jazyk, který je podobný JavaScriptu. Uživatelé Unity a často i samotný tým Unity ho nazývají UnityScript. Pro práci v Unity je lepší používat spíše C#, a to právě z toho důvodu, že se v Unity používá stejně jako kdekoli jinde. Díky tomu je snazší najít nápovědu při snaze vyřešit problém, se kterým si uživatel neví rady. (21)

Mezi hry vytvořené v tomto enginu lze zařadit Pokémon Go, Monument Valley, Call Of Duty: Mobile, Beat Saber nebo Cuphead. (23), (22)

Výhody Unity:

- Nejrychlejší herní engine na trhu.
- Herní enginy Unity jsou široce využívány díky podpoře pro 27 platform.

- Vytvořenou aplikaci lze jednoduše sdílet mezi PC, mobilním zařízením a webem.
- Kvalitní zvukové a vizuální efekty výrazně usnadňují vývoj hry a zlepšují celkový herní zážitek.
- Dostupná velice srozumitelná dokumentace.
- Unity umožňuje snadné ladění a debugování. (22)

3.2.2 Unreal Engine

Unreal Engine je populární zejména díky uživatelsky přívětivým funkcím, mohou ho používat úplní začátečníci, protože nevyžaduje žádné programátorské dovednosti. Hru je tak možné upravovat bez zásahu do kódu. Další výhodou je, že Unreal Engine nabízí možnost testovat hru přímo v aplikaci. Stejně jako Unity, Unreal Engine podporuje více platform najednou. (22)

Dále se používá ve filmovém průmyslu ke tvorbě animovaných filmů a scén, nabízí totiž možnost tvorby krásné 3D grafiky. (23)

Mezi známé hry vytvořené v tomto enginu patří například Injustice 2 nebo Mortal Kombat. (22)

Výhody, které nabízí Unreal Engine:

- Zdarma a open-source software
- Multiplatformní herní engine
- Nejlepší nástroj pro optimalizaci her
- Možnost využití předem připravených prostředků k dosažení fotorealistických obrazů pro hru. (22)

3.2.3 Cryengine

Cryengine je vhodným nástrojem pro tvorbu her s dobrými vizuálními prvky. Nabízí řadu nástrojů, které pomáhají v tvorbě pestrého prostředí hry. Jedním z těchto nástrojů je například Flowgraph. Dále nabízí nástroje pro začátečníky, které slouží pro lepší a snazší pochopení samotného enginu. Mezi nejznámější hry vytvořené pomocí tohoto enginu lze zařadit Farcry, Warface nebo Arena of Fate. (22)

Výhody Cryengine:

- Umožňuje vyvinout skvělé herní prostředí.
- Disponuje nejvýkonějším zvukovým nástrojem Fmod.

- Dokáže si poradit s náročným a bohatým prostředím hry.
- Poskytuje rozsáhlou podporu vývojářům. (22)

3.2.4 Solar2D (Corona SDK)

Solar2D známý také jako Corona SDK se využívá především ke tvorbě 2D mobilních her. Využívá skriptovací jazyk Lua, který je možné se relativně snadno naučit, a je dostupný zdarma. Solar2D nabízí pro uživatele tržiště, ve kterém jsou dostupné různé pluginy nebo funkce. Má také poměrně aktivní komunitu a podrobnou dokumentaci. Navíc má také Solar2D Playground, což je interaktivní webová stránka, díky které je možné spouštět a vytvářet projekty online. Mezi hry, které byly vytvořené v tomto enginu, patří například Flappy Bird nebo Match 3 Space RPG.

Výhody Corona SDK:

- Software, který využívá jediný kód pro vývoj her, což eliminuje potřebu vytvářet oddělené kódy pro různé platformy.
- Podporuje snadnou monetizaci díky zabudované podpoře pro bannerové reklamy, což usnadňuje generování příjmů.
- Corona dosahuje vysokého počtu snímků za sekundu, což zaručuje plynulý a konzistentní herní zážitek.
- Vývoj v Corona SDK připomíná spíše práci designéra než vývojáře, díky možnosti integrovat grafický obsah. (22)

3.2.5 SpriteKit

SpriteKit byl vyvinut společností Apple a je tedy jedním z nejlepších způsobů, jak vyvíjet hry přímo pro iOS zařízení. Lze pomocí něho vytvářet dvourozměrnou grafiku a je považován za uživatelsky přívětivý software na vytváření her či graficky náročných aplikací. (23)

SpriteKit podporuje dva jazyky, a to Swift a Objective-C. Hry vytvořené v tomto prostředí jsou například Sumo pop nebo 10*10 Blocks. (22)

Výhody SpriteKit:

- Vyvíjen od společnosti Apple, proto je přesný a ideální pro vývoj aplikací pro iOS.
- Dobrá spolupráce pomocí aplikace Xcode při vkládání obrázků.

- Framework disponuje nástroji, které umožňují simulovat fyzikální interakci v herním prostředí. (22)

3.2.6 Cocos2D

Jde o open-source engine, který slouží ke tvorbě her na více platform. Primárně je zaměřený ke tvorbě 2D a 3D mobilních her a je ideální volbou pro pokročilé vývojáře. Podporuje několik jazyků, mezi které patří C++, Lua nebo JavaScript. (23)

Má podrobnou dokumentaci a také velkou komunitu, díky čemuž je možné se poměrně snadno s nástrojem naučit pracovat. Nabízí také editor, který celý proces vývoje hry usnadňuje, a to včetně odstranění chyb, náhledů hry nebo publikaci na ostatních platformách. (22)

Výhody Cocos2D-X

- Open-source engine.
- Podporuje vývoj multiplatformních her.
- Malá velikost herních souborů, pouhých 1,5 MB. (22)

3.2.7 BuildBox

BuildBox se využívá pro tvorbu dvourozměrných a trojrozměrných her bez nutnosti programování a je tak vhodný pro začátečníky nikoli experty. Díky tomuto enginu lze vyvinout hry poměrně snadno a rychle, celý proces vytváření hry spočívá ve vytváření obrázků, kterým se přiřazují určité role, a to je v podstatě vše. Pokud jsou v aplikaci prováděny nějaké změny dají se provést přímo v samotném enginu, kde se také může daná změna otestovat. (22)

Výhody BuildBox:

- BuildBox nabízí velké množství zvukových efektů a animací.
- Snadná implementace reklamy do her.
- K dispozici herní šablony, které usnadňují vývoj.
- Nejnovější verze umožňuje vývoj 3D her bez nutnosti psaní kódu. (22)

3.2.8 Construct 3

Jedná o herní engine, který je perfektně vhodný pro úplné začátečníky. Byl vyvinutý společností Scirra a využívá jazyka HTML5, který je přesně vhodný pro ne příliš zdatné

uživatelé v oblasti programování. Podporuje, ale také JavaScript a dále využívá virtuální skriptovací systém, který celou práci v samotném nástroji ještě více usnadňuje. (22)

Výhody Construct 3:

- Vynikající pro vývoj 2D her.
- Vývoj bez znalosti programování.
- Cenově dostupný software. (22)

3.2.9 GameMaker Studio 2

Tento herní engine byl vytvořen společností Yoyo games a slouží pro tvorbu dvourozměrných her. Využívá tzv. „drag-and-drop“ funkce, díky čemuž je možné vytvořit hru snadno a rychle. K tomu také přispívá samotný jazyk tohoto nástroje, který není těžký se naučit. GameMaker Studio 2 je cenově přívětivý, a tak není třeba utrácet velké částky za práci v tomto nástroji. Umožňuje uživatelům také změnu pracovního prostředí nástroje, a tak se jedná i o uživatelsky přívětivý nástroj pro tvorbu her. Jednou z her, která byla vytvořena v tomto nástroji je Undertale. (22)

Výhody GameMaker Studio 2:

- Skvělý herní engine pro začátečníky.
- Možnost vytvářet hry bez programování.
- Snadný export na jiné platformy. (22)

3.2.10 MonoGame

MonoGame je nástroj, který slouží ke tvorbě 2D her na více platformech. Je založen na frameworku XNA od společnosti Microsoft. Využívá dva programovací jazyky, C# a .NET. Výhodami tohoto engine je jeho velká komunita uživatelů a dostupnost zdarma. (23)

Výhody MonoGame:

- Vysoce přizpůsobitelný engine pro vývoj her.
- Skvělý engine pro vývoj 2D her.
- Open-source framework. (22)

3.2.11 Haxe

Jedná se o další open-source nástroj pro tvorbu her na více platformech. Je nabízen zdarma, přičemž za určité poplatky lze získat nějakou formu podpory.

Jako programovací jazyk se podobá jazykům jako C++ nebo PHP. Disponuje mnoha frameworky a knihovnamy, které dále slouží pro vývoj her nebo přizpůsobení hry na ostatní platformy. (22), (23)

Výhody Haxe:

- Snadno použitelný se znalostí JavaScriptu.
- Kompilátor Haxe je velice rychlý. (22)

3.2.12 Gideros

Gideros umožňuje tvorbu dvourozměrných a trojrozměrných her, je zdarma a open-source. Je populární díky snadnému a rychlému vývoji her. Nabízí open-source pluginy a sady vývojových nástrojů. (23)

Výhody Gideros:

- Bezplatná technologie.
- Možnost okamžitého testování hry. (22)

3.2.13 Godot

Godot je další herní engine, který umožňuje tvorbu jak 2D, tak 3D her pro různé platformy. Populární je zejména při tvorbě 2D her, protože využívá systém měření, který je založený na pixelech. Uživatelům nabízí velkou sadu nástrojů, která jim pomáhá při vývoji her a také jim nabízí úpravu uživatelského rozhraní. Usnadňuje tak tvorbu animátorům, návrhářům úrovní a umělcům. (23)

Výhody Godot:

- Godot podporuje širokou škálu programovacích jazyků a jejich kombinaci.
- Dostupný zdarma.
- Veškerý obsah vytvořený pomocí tohoto herního enginu patří jenom jeho autorovi. (22)

3.2.14 Další herní enginy

Mezi další herní enginy patří například Phaser, Defold, GameSalad, Marmalade, AppGameKit, Starling. (22)

3.3 Proces vývoje mobilní hry

Proces vývoje mobilní hry se velice podobá vývoji mobilních aplikací, i přesto je v některých částech jiný. Také se velice liší od vývoje hry pro ostatní platformy jako počítače a konzole. Pro představu, vývoj hry pro mobilní zařízení je podstatně rychlejší a není tak časově náročný oproti vývoji her pro ostatní platformy.

Vývoj mobilní hry by měl zahrnovat nějaké plánování, hledání nápadu, jak by hra měla vypadat. Potom by se měl vybrat vhodný nástroj pomocí kterého bude samotná hra vznikat. V procesu vývoje hry by neměla chybět tvorba dokumentu, který by hru do detailu popisoval. Proces by měl dále zahrnovat volbu monetizační strategie. Potom by se mělo přejít na navrhování herních struktur a wireframů. Dále nastává samotný vývoj hry a poté testování. Poslední krok by měl zahrnovat publikaci hry a případné updaty.

3.3.1 Nápad a plán na vývoj hry

Vývoj mobilní hry není tak finančně a časově nákladný jako u her pro jiné platformy. I přesto by celý proces měl mít určitou formu plánování, zejména pokud se jedná velký komerční projekt.

První krok plánování by tedy měl spočívat v definici co se vlastně bude vytvářet za hru a pro koho by hra měla být určena a určitě je vhodné nejlepší nápady sepsat. (24)

Vývojáři by si při plánování měli klást tyto otázky:

- Jaký typ hry se pokoušíte vyvinout, včetně zaměření, žánru a jakou bude mít hra grafiku, 2D nebo 3D?
- Na koho se hra nejvíce zaměřuje?
- Je plánem hru zpeněžit?
- Jaké máte stanové cíle? Pokoušíte se cílit na největší počet hráčů, maximalizovat zisk, nebo si splnit osobní sen?
- Jaký je finanční plán a plánované časové období pro vývoj?
- Kdo je hlavním konkurentem na trhu? (24)

Zapisování a plánování usnadňuje celý proces, a proto je to důležitá část při tvorbě hry. (24)

3.3.1.1 Atributy dobré hry

Mezi nejpoblárnější hry často patří ty s jednoduchým ovládáním. Hra na mobilní zařízení by tedy měla být jednoduchá, měla by být snadná na pochopení a zároveň by neměla mít

příliš dlouhé úrovně. Je to z toho důvodu, že mobilní uživatelé často nemají příliš mnoho času, takže vytvářet složitá pravidla, proto není úplně rozumné.

Další faktory, na které by se měl brát ohled při vývoji her jsou jakým způsobem bude uživatel hru ovládat nebo jak bude využívat funkce moderních mobilních zařízení. Hra by měla mít co nejjednodušší schéma pro ovládání tak, aby poskytovala hráči přímou interakci s prvky dané hry. Pro hry na mobilní zařízení není vhodné vytvářet jakési ovládání podobné ovladači pro konzole. Dotykové ovládání je velice užitečnou funkcionalitou, kterou by hra měla mít. Díky tomu, že bude mít uživatel možnost přímo interagovat s prvky hry, bude lépe vtažen do hry. Dobrým příkladem jsou deskové hry například šachy, ve kterých hráč může pohybovat figurkou pomocí dotykové obrazovky, aniž by musel používat nějaký obrazovkový joystick.

Vývojář by měl hru navrhovat tak, aby byla zábavná, je to ta nejdůležitější úvaha při tvorbě hry. Každý vidí zábavu v něčem jiném, proto je nejlepší, pokud je hra testována před vydáním, tak se vývojář dozví, jak by hru mohl upravit. Hra by také uživateli měla přinášet nějakou formu výzvy. Neměla by být příliš jednoduchá a ani příliš těžká, to by mohlo hráče rychle odradit od hraní. Pokud hráč dosáhne nějakého milníku ve hře, měl by být nějakou formou odměněn. Odměna může představovat například nějakou vtipnou animaci. Důležité jsou také herní statistiky a žebříčky, kterými lze podněcovat soutěživost mezi hráči.

Dále by hra měla mít poutavou grafiku. Důležitý je zejména design ikonky, protože to je první věc, kterou uživatel uvidí a která na něj může zapůsobit nebo naopak ho může od stažení aplikace odradit. Pěkná grafika má tedy velký vliv na to, zda bude hra úspěšná nebo ne.(25)

3.3.2 Vhodná platforma a engine

Po plánování je důležité vybrat vhodný engine, přes který bude hra vytvářena. Hru lze vytvořit na jednu nebo více platformách najednou. Mohou být vytvořeny pro Android nebo iOS, což jsou operační systémy pro mobilní zařízení. Hry je možné dále vyvíjet pro konzole nebo počítače. Mezi počítačové operační systémy patří Windows, macOS a Linux. (24)

Každý operační systém funguje trochu jiným způsobem, a proto je důležité vybrat správný nástroj, který podporuje daný operační systém. Mezi další prvky, na které je třeba brát ohled při výběru nástroje jsou také jeho schopnosti. Některé enginy totiž umožňují tvorbu 2D her a jiné zase 3D her. Dále je třeba brát ohled na složitost celého nástroje, výkon nebo také cenu. Pro tvorbu jednoduché 2D hry je vhodné vybrat například Unity nebo Godot.

Naopak pro tvorbu složitější a rozsáhlejší 3D hry je možné zařadit herní enginy jako Unreal Engine. (26)

3.3.3 Rozdíly mezi vývojem her pro mobilní zařízení a počítače

Vývoj her pro mobilní zařízení a počítače představuje dva odlišné přístupy, každý s vlastními výzvami. Hlavním rozdílem je využití hardwarových zdrojů a přístup k designu her.

- **Hardwarová omezení** – Mobilní hry jsou vyvíjeny s ohledem na omezený výkon a úložný prostor mobilních zařízení, což znamená, že musí být optimalizované pro slabý hardware. Naproti tomu vývojáři PC her mají možnost využívat pokročilého hardwaru, což umožňuje vytváření graficky náročnějších a složitějších her.
- **Složitost a grafika** – PC hry často nabízejí složitější herní mechanismy a lepší grafiku, zatímco mobilní hry se obvykle zaměřují na jednoduchost. PC hry mohou využívat komplexních ovládacích prvků, jako jsou klávesnice a myši, zatímco mobilní hry jsou omezeny na dotykové obrazovky.
- **Vývojové nástroje a náklady** – Vývoj mobilních her může být rychlejší a méně nákladný, často s možností využití bezplatných nástrojů a knihoven. Naproti tomu vývoj PC her vyžaduje často dražší software a hardwarové vybavení, což vede k vyšším počátečním investicím a delší časové náročnosti.
- **Tržní potenciál a dostupnost** – Mobilní hry mohou dosáhnout širšího publika díky vysokému zastoupení chytrých telefonů a tabletů. PC hry zase mohou cílit na specifické segmenty hráčů, kteří vyhledávají ohromující herní zážitek. Liší se také v dostupnosti. Mobilní hry jsou často dostupné prostřednictvím obchodů s aplikacemi, zatímco PC hry mohou být distribuovány digitálně nebo fyzicky.
- **Uživatelé** – Mobilní hry jsou populární mezi příležitostnými hráči díky snadné dostupnosti a jednoduchosti. PC hry oslovují jak především hardcore hráče.

Vývoj počítačové hry je skvělou volbou, pokud je cílem vytvořit hru s užasnou grafikou a hrátelností. Pokud je prioritou oslovit široké a nenáročné publikum a zároveň vytvořit hru rychle a levně, pak je tvorba mobilní hry vhodná volba. (27)

3.3.4 Game design dokument

„Game design dokument je již řadu let standardem v herním průmyslu, pomáhá týmům s vytyčením cílů v projektu a jejich plněním. Správně vytvořený dokument nasměruje vývoj hry správným směrem a umožní se všem členům týmu zorientovat v projektu.“ (28)

Je to dokument, který obsahuje přehled veškerých funkcí požadavků samotné hry. V tomto dokumentu jsou tedy zaznamenané všechny kroky, které byly provedeny v průběhu vytváření hry. (24)

Dokument je důležitý zejména pokud se jedná hru větších rozměrů. Na takové hře většinou pracuje velký tým lidí, a tak je pro vysvětlení nápadů či změn ve hře důležité vytvořit dokument, který hru popisuje. Také je možné, že team v průběhu vývoje někdo opustí nebo naopak někdo nový přibude, a proto je důležité, aby byla hra zdokumentována. Mimo to by tento dokument mohl posloužit také jako prezentace investorům. (28)

Samotný dokument se v průběhu práce na projektu mění a přizpůsobuje se potřebám. Jak by daný dokument měl vypadat není přesně dáno, a tak se mohou lišit zvláště podle toho o jak velký projekt se jedná. V dokumentu se nejčastěji objevují kapitoly jako: *„úvod, game design, game play, grafika, modely, level design, příběh, charaktery, user interface (UI), ovládání, zvuk, distribuce a marketing a určení cílového publika.“ (28)* V dokumentu také bývá uvedený seznam lidí, kteří na projektu pracují a časový rámec celého projektu. Mohou tam být také obrázky, grafy nebo prototyp samotné hry.

Pro vývoj malé hry není třeba vytvářet rozsáhlý dokument, ale určitě i v tomto případě by nějaký měl být. Mohl by se například skládat z těchto kapitol: (28)

- **Filozofie hry:** *krátký popis hry, který vystihne očekávané zkušenosti uživatele s hrou.*
- **Souhrn:** *v několika větách popsané základní informace o hře.*
- **Charakteristika:** *popis herních mechanik a dalších charakteristických prvků pro hru.*
- **Cílové publikum:** *definice publika, které bude s největší pravděpodobností hru hrát.*
- **Game design:** *detailní popsání herních mechanismů, většinou se tento bod skládá z více pod bodů (příběh, prostředí, multiplayer).*
- **UI:** *popis ovládání hry (klávesnice, myš, ovladač).*

- **Umělecký styl:** definice zvoleného uměleckého stylu (*pixel art, vektor, realistické 3D, low polygon*).
- **Zvuk:** popis toho jakou hudbu a zvuky bude hra využívat.
- **Časový plán:** stručně načrtnutý časový rámeček projektu.“ (28)

3.3.5 Monetizace

Monetizace je důležitým krokem při tvorbě hry a měl by se naplánovat hned ze začátku projektu. Zvolení správné strategie, jak monetizovat hru je důležité, protože ovlivňuje funkčnost a mechaniku hry. Mezi modely monetizace můžeme zařadit reklamy, nákupy ve hře nebo balíčky podobné předplatnému. (24)

3.3.6 Design hry a wireframů

V tomto kroku figurují převážně grafici, herní designéři a umělci, kteří mají za úkol tvorbu hry po grafické stránce. Musejí celý nápad převést do grafické podoby, je potřeba vytvořit návrh struktury podle, kterého se bude tvořit samotná hra. Návrh hry by měl zahrnovat tyto pod úkoly:

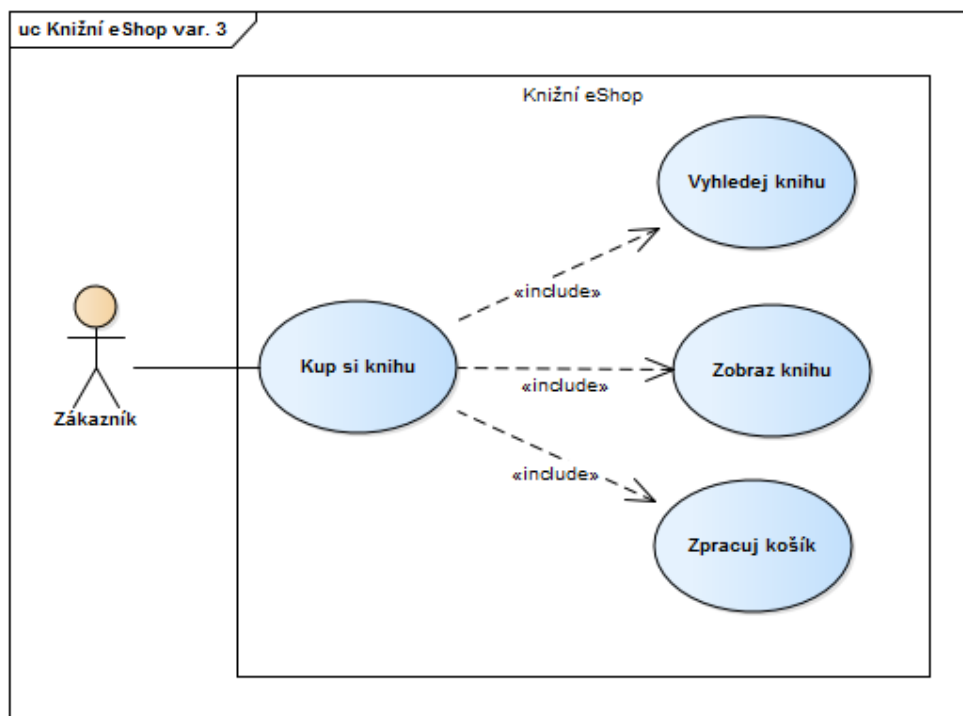
- Vypracovaný návrh příběhu a konceptu.
- Plánování úrovní hry.
- Vývoj herních pravidel a funkcionalit.
- Vytváření postav a herních assetů.
- Zavedení způsobů monetizace hry.
- Konstrukce hlavních a sekundárních cílů a událostí. (24)

Tento proces také zahrnuje tvorbu wireframů, které zobrazují, jaké hráč může provést kroky a co bude následovat, pokud daný krok nastane. S těmito wireframy potom pracuje team vývojářů. (24)

3.3.6.1 Use-case diagram

Use-case diagram poskytuje přehled o tom, kdo používá systém a jak s ním tyto osoby interagují. Osobám se v tomto případě říká aktéři. Pro tvorbu use-case diagramů se využívá předem definovaná sada symbolů a propojení. Nástroj umožňuje týmu vizualizovat různé situace, ve kterých systém nebo aplikace interaguje s lidmi. Dokáže reprezentovat cíle, kterých se uživatelé snaží dosáhnout při používání aplikace. (29)

V diagramu každý use-case zobrazuje specifický scénář, který říká, jak se systémem komunikuje uživatel. Use-case umožňuje zobrazení požadavku na nový systém. Typy požadavků jsou dva, funkční a nefunkční. Funkční požadavek vyjadřuje, jak se systém zachová při zadání vstupních podmínek. Nefunkční požadavek představuje, jak dobře systém pracuje. (30)



Obrázek 4: Příklad use-diagramu (31)

Tyto diagramy nezasahují do hloubky jednotlivých procesů. Spíše poskytují obecný přehled o vztazích mezi různými prvky, jako jsou use-case, aktéři zapojení do procesu a samotný systém. Jsou často doporučovány jako lepší popis složitějších situací.

Pro tvorbu diagramů se používá sada nástrojů Unified Modeling Language (UML), kde jsou use-case znázorněny oválnými formami a aktéři jsou reprezentováni figurkami. Vztahy mezi aktéry a use-case jsou zobrazeny čarami. Hranice systému lze naznačit obkreslením všech use-case a aktérů do rámečku. (29)

Use-case diagramy jsou vhodné pro:

- Zobrazení cílů mezi systémem a uživatelem.
- Definici a strukturování funkčních požadavků systému.
- Specifikaci kontextu systému a jeho požadavků.
- Modelování základního průběhu akcí v daném scénáři. (29)

Pro pochopení, co obsahuje Use-case diagram, je klíčové znát jeho základní prvky:

- **Aktéři** – Ti představují uživatele nebo skupiny, které s daným systémem komunikují. Mohou to být jednotlivci, celé organizace nebo dokonce jiné systémy. Aktéři jsou vždy externí prvky vůči systému.
- **Systém** – Jedná se o soubor činností a interakcí mezi systémem a jeho aktéry. Tento pojem může být rovněž vykládán jako specifický scénář.
- **Cíle** – Ty reprezentují zamýšlené výsledky, kterých má být dosaženo prostřednictvím různých případů užití. (29)

Notace v use-case diagramu je poměrně přímá a neobsahuje tak rozmanité typy symbolů, jako je tomu u jiných druhů UML diagramů. Zde je přehled základních prvků, které lze použít v use-case diagramu:

- **Use-case** – Neboli případy užití, jsou znázorněni pomocí oválů a reprezentují jednotlivé akce nebo činnosti, které mohou uživatelé se systémem provádět.
- **Aktéři** – Jsou zobrazeni jako postavičky a symbolizují osoby nebo systémy, které s případy užití interagují.
- **Asociace** – Tyto spojnice ukazují na vztahy mezi aktéry a případy užití, ukazují, kdo může, s jakými případy užití interagovat.
- **Hraniční rámeček systému** – Rámeček, který definuje rozsah systému, určuje, které případy patří do systému a které jsou mimo jeho rozsah.
- **Balíčky** – Tyto UML tvary umožňují organizovat prvky diagramu do skupin. Díky tomu se zjednodušuje orientace v diagramu. (29)

3.3.6.2 Vývojový diagram

Před začátkem psaní kódu je vhodné promyslet si řešení úlohy, lépe řečeno provést algoritmizaci dané úlohy. Algoritmus je definován jako série kroků, které vedou k řešení daného problému. Algoritmus je spojený s programováním, přičemž před programování je vhodné úlohu rozdělit na jednotlivé kroky. Ke grafickému znázornění jednotlivých kroků algoritmu se používá vývojový diagram. Používají se v něm specifické symboly spojené orientovanými šipkami, které ukazují postup od začátku k výsledku. Tato metoda umožňuje jednoduše zobrazit a pochopit postup řešení algoritmu. Je to základní krok v procesu programování, který výrazně zefektivní vývoj a pomůže k řešení problému. (32)

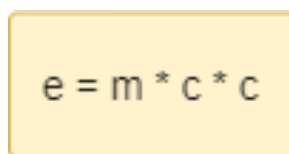
Vývojové diagramy jsou tvořeny pomocí grafických symbolů, které mají různé tvary a slouží k reprezentaci různých operací nebo situací v programu. Každý symbol může obsahovat detailní informaci k lepšímu pochopení dané části algoritmu. (33)

Diagramy obvykle začínají a často i končí u speciálního symbolu nazývaného Terminál. Terminál bývá zobrazen jako obdélník, ale může mít i tvar elipsy nebo kruhu. Na začátku diagramu nese terminál popisek „Start“ a na konci „Konec“. Existují také diagramy, které mají více koncových bodů. (34)



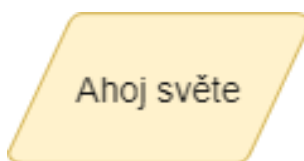
Obrázek 5: Terminál (34)

Kroky algoritmu nebo procesu jsou typicky znázorněny pomocí obdélníků, existují však výjimky pro určité speciální symboly, jako je terminál nebo jiné. V různých diagramech mohou mít tyto procesy hranaté nebo naopak zaoblené rohy. Důležité je, aby byly jasně odlišitelné od terminálu, který má výrazně zaoblené rohy.



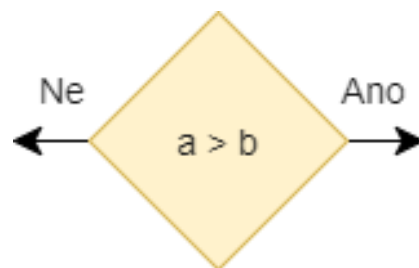
Obrázek 6: Krok algoritmu (34)

Většina programů nějakým způsobem interaguje s daty. Pro znázornění těchto procesů se často používá tvar kosodélníku. Tento symbol typicky reprezentuje načítání dat od uživatele a jejich ukládání do proměnných, nebo naopak výpis dat zpět uživateli. I když existují i jiné specifické symboly pro záznam nebo výpis dat, kosodélník je v mnoha případech preferovaným symbolem pro tyto operace. V nějakém diagramu by mohl kosodélník obsahovat například instrukci pro výpis zprávy „Ahoj světe“.



Obrázek 7: Výpis zprávy (34)

Dalším klíčovým symbolem pro tvorbu diagramů je kosočtverec, který symbolizuje rozhodovací bod neboli podmínku. Tento prvek se využívá k reprezentaci míst, kde program musí rozhodnout, kterou cestou se vydat na základě splnění nebo nesplnění určité podmínky. Cykly lze znázornit pomocí podmínek tak, že jeden z výstupů z rozhodovacího bodu vede zpět na jeho začátek čímž vzniká smyčka. (34)



Obrázek 8: Podmínka (34)

3.3.7 Vývoj hry

Samotný proces vývoje hry většinou zahrnuje několik kroků, kterými tato část prochází. Mezi tyto kroky patří určitě tvorba uměleckých prototypů. Dále tvorba zvukového designu. Poté se vytváření demoverze hry a potom probíhá testování. Poslední krok se zaměřuje na opravu chyb a provádění patřičných změn. (24)

Celý vývoj hry probíhá v předem vybraném nástroji, tímto nástrojem může být například Unity. V něm se celý projekt nastavuje, programuje se ovládání pomocí jazyka C#, pracuje se kamerou a vytvářejí se úrovně hry. Pomocí tohoto nástroje lze také realizovat monetizační strategii. Celý vývoj probíhá pro předem stanovená zařízení, pro které je hra tvořena. (35)

3.3.7.1 Programovací jazyk

Programovací jazyk je sada pravidel určená pro instruování počítače k vykonání specifických úkolů pomocí algoritmů. Tento jazyk slouží ke komunikaci mezi vývojářem a počítačovým hardwarem. Je to jazyk, který umožňuje programátorům psát programy, které určují, jaké operace má počítač vykonat. (36)

Je to nástroj pro psaní kódu a vývoj počítačových programů, aplikací či webových stránek. Existují stovky programovacích jazyků, jejichž počet se stále mění s pokrokem technologií a specifickými potřebami různých oborů. Každý programovací jazyk má své unikátní vlastnosti a pravidla, ale základní principy programování jsou společné pro všechny. (37)

3.3.7.2 Jazyk C#

C# je moderní programovací jazyk, navržený pro široké využití v různých oblastech. Tento jazyk se především uplatňuje v rámci frameworku Windows .NET. Mimo to je také populární na open-source platformách. Patří do skupiny objektově orientovaných

programovacích jazyků. Jazyk je to poměrně nový i přesto je považovaný za velice spolehlivý.

Tento jazyk je od společnosti Microsoft a je určený pro širokou škálu vývoje, včetně mobilních a desktopových aplikací, webových stránek a her. Zásadní role C# je ve vývoji Windows aplikací díky integraci s .NET frameworkem. Dále se používá pro vývoj her v Unity, kde se stává preferovaným jazykem pro třetinu nejlepších her. Vzhledem k tomu, že se jedná o objektově orientovaný programovací jazyk tak je vhodný také pro tvorbu webových stránek.

C# získal popularitu v herním vývoji hlavně díky tomu, že je to vysokoúrovňový jazyk, který nabízí jednodušší syntaxi oproti nižší úrovni jazyků jako jsou C++ či Assembly. To usnadňuje programátorům psaní a údržbu kódu, což jim ušetří čas.

Tento jazyk vyniká také díky své knihovně tříd, která má mnoho před vytvořených funkcí a kódů, jenž usnadňují tvorbu komplexních herních systémů. Dále nabízí automatické řízení paměti, což ulehčuje vývojářům práci, protože se nemusí starat o alokaci a dealokaci paměti. Tuhle úlohu má na starosti integrovaný garbage collector.

Jednou z největších předností C# v herním vývoji je jeho schopnost fungovat na různých platformách. Programátoři mohou psát kód, který je poté možné spustit na mnoha operačních systémech, jako je Linux, Windows, macOS, dokonce i na mobilních zařízeních s Androidem nebo iOS, anebo také ve webových prohlížečích. Tato vlastnost umožňuje tvůrcům her oslovit širší publikum bez potřeby psaní specifického kódu pro každou platformu.

C# se využívá v mnoha oblastech herního vývoje. Používá se k tvorbě enginů, vývojových nástrojů a ke skriptování. Herní engine jako Unity ho využívá pro tvorbu herní logiky a uživatelských rozhraní. Využívá se také při tvorbě speciálních vývojových nástrojů, jako jsou editory úrovní. Dále se využívá pro psaní herních skriptů. Skript je malá část kódu, která se spouští vždy v určitý moment hry. (38)

Použití jazyka C# k vývoji her má tyto výhody:

- **Lepší vývoj** – C# usnadňuje psaní kódu, urychluje celý vývoj, což může vést ke snížení nákladů
- **Kompatibilita** – Jazyk lze zkompileovat, aby byl kompatibilní s více platformami. To zjednodušuje tvorbu hry a umožňuje hru přesunout i na jiné zařízení či operační systémy.

- **Správa paměti** – Jazyk automaticky spravuje paměť, díky čemuž je možné minimalizovat riziko uvolnění paměti.
- **Ladění** – Lépe si poradí s identifikací a ladění chyb v kódu.
- **Komunita** – Tento jazyk je podporován rozsáhlou komunitou vývojářů, kteří sdílejí své znalosti, což vede ke snazšímu učení. (38)

3.3.8 Nahrání hry do obchodu a tvorba aktualizací

Posledním krokem celého procesu vývoje mobilní hry je nahrát ji do obchodů obou platforem, ze kterých si uživatelé mohou hru stáhnout. Soubor se hrou pro Android lze také nahrát na webovou stránku studia, které na hře pracovalo. Obchody, na které je možné hru nahrát jsou dva, Google play a App store. Každý z těchto obchodů má určitá pravidla, které je pro publikaci hry třeba dodržovat.

Při nahrávání hry do obchodů by se nemělo zapomenout na popis a prezentaci hry. Obrázky nebo videa, která budou uvedena v obchodech by měla být poutavá tak, aby lákala uživatele ke stažení. Dále je třeba vybrat vhodná klíčová slova, aby byla snáze vyhledatelná v obchodě. Hra by také měla dostávat pravidelné aktualizace, které do hry budou přidávat nějaký obsah nebo hře opravovat chyby. Nový obsah by se měl přidávat alespoň každý měsíc nebo lépe provést aktualizaci jednou týdně nebo jednou za dva týdny. Je to důležité zejména z toho důvodu, protože hra by mohla čelit riziku rychlého vyřazení. (24)

4 Tvorba mobilní hry

4.1 Plán hry

Pro vytvoření hry je potřeba si celý proces naplánovat. Ze všeho nejdříve je tedy důležité určit jaký cíl by hra měla mít, kdo by měl být uživatelem hry a jak by měla vypadat a fungovat. Hra by měla být vzdělávacího charakteru, která je vhodná pro děti ve školce a žáky prvního stupně základní školy. Uživatel si pomocí aplikace bude zlepšovat své matematické dovednosti pomocí počítání jednoduchých matematických příkladů.

Celý průběh hry bude spočívat v tom, že se uživateli zobrazí náhodný matematický příklad a on bude mít za úkol přijít na výsledek. Hra uživateli nabídne určitý počet výsledků, ze kterých bude vybírat. Na základě toho, jaký uživatel zvolí výsledek bude aplikace dále reagovat. Pokud hráč vybere špatně, aplikace mu strhne pokus na vyřešení daného příkladu. Pokud hráč odpoví správně, dostane bod za správné vyřešení příkladu. Hráč bude dále limitován časem, který bude mít na správné vyřešení příkladu. Pokud hráči dojde čas, hra mu za to strhne pokus. Pokud uživateli všechny pokusy na vyřešení příkladu dojdou, celá hra je u konce. Po konci hry bude hráč informován o tom kolik bodů získal tedy kolik příkladů správně vyřešil.

Hra by měla být převážně jednoduchá, neměla by mít žádné složité prvky, které by se uživatel musel učit. Díky tomu bude mít také jednoduché ovládání. Celá hra se bude ovládat pomocí dotykové obrazovky mobilního zařízení. Vzhledem k tomu, jak je hra navržena by měla uživatelům přinášet určitou výzvu. Díky tomu budou splněny atributy, které by měla mít každá dobrá hra.

4.2 Funkční a nefunkční požadavky

Na úvod byl vytvořený seznam funkčních a nefunkčních požadavků pro hru. Dané požadavky jsou popsány v následující tabulce.

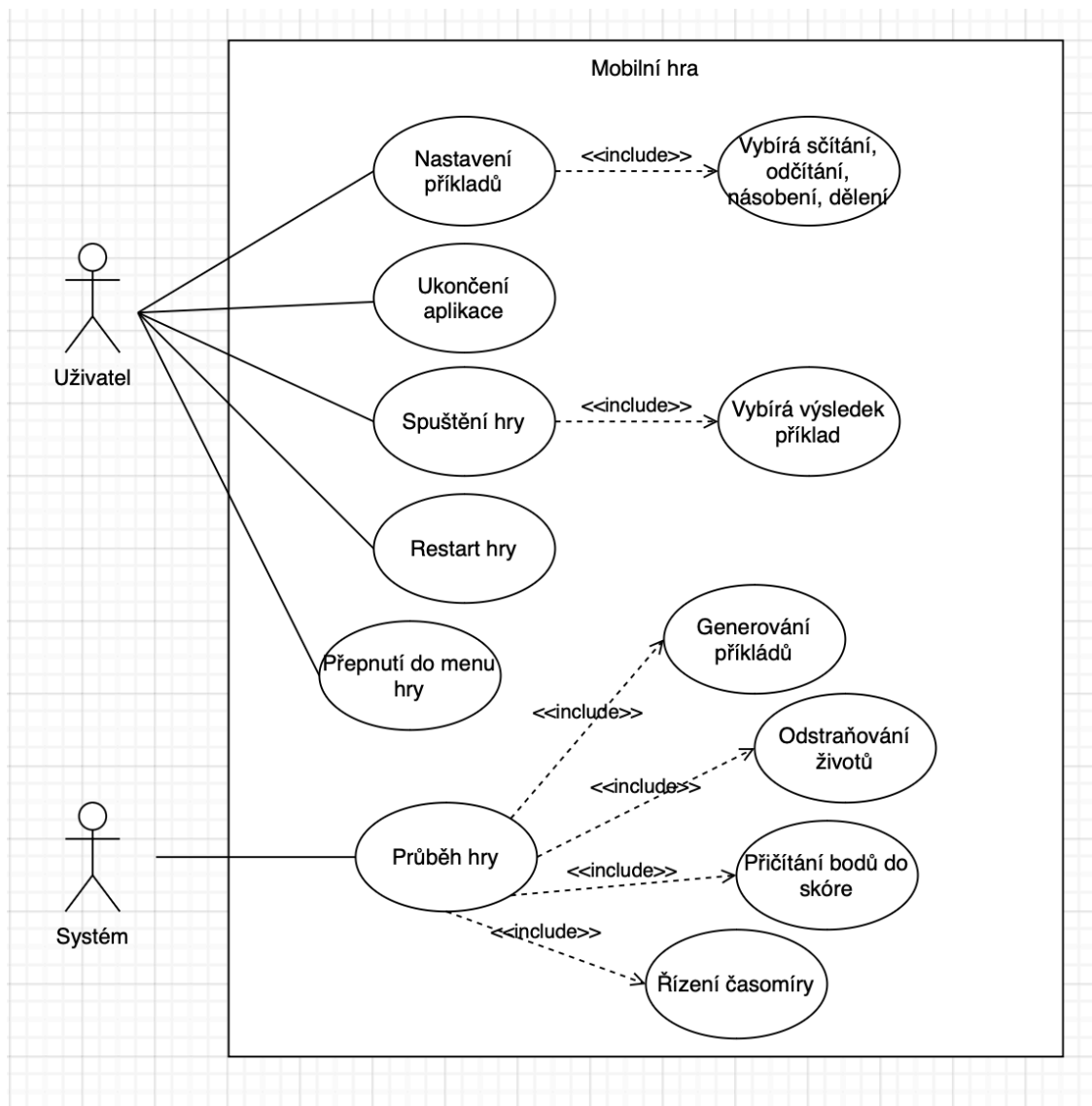
Funkční požadavky		
ID	Název	Popis
1	Nastavení	Uživatel si před hraním může vybrat, které matematické příklady se budou generovat.
2	Start hry	Uživatel může odstartovat hru pomocí tlačítka „Start hry“.

3	Ukončení aplikace	Uživatel může ukončit aplikaci stisknutím tlačítka „Konec hry“.
4	Generování příkladů	Systém generuje matematické příklady na základě toho, jaké uživatel nastavil v nastavení.
5	Vybírá výsledek	Uživatel vybírá výsledek příkladu.
6	Získání bodu	Systém přičte bod do skóre, pokud hráč odpoví správně
7	Ztráta života	Systém odstraní život, pokud hráč odpoví špatně.
8	Časový limit	Uživatel má časový limit na určení výsledku matematického příkladu.
9	Konec hry	Uživatel má možnost vidět své finální skóre, pokud mu dojdou všechny životy.
10	Restart hry	Uživatel má možnost restartovat hru a začít znovu.
11	Vrácení do menu	Uživatel má možnost vrátit se zpět do hlavního menu.
Nefunkční požadavky		
12	Odezva	Odezva na uživatelské interakce musí být rychlá a plynulá.
13	Uživatelské rozhraní	Uživatelské rozhraní musí být intuitivní a snadno použitelné

Tabulka 1: Funkční a nefunkční požadavky

4.3 Use-case diagram

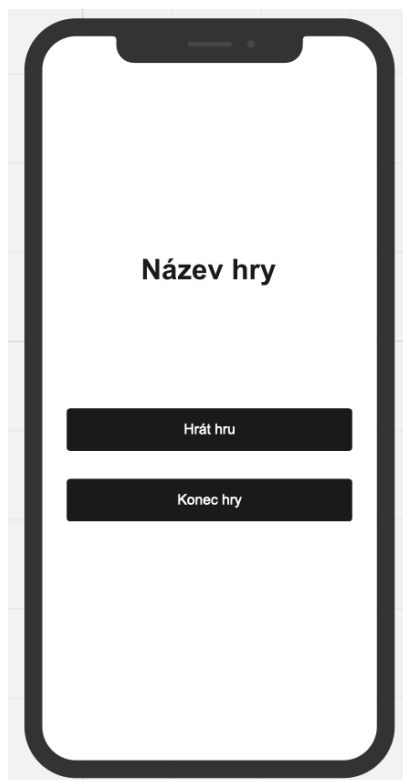
Na základě požadavků byl také vytvořený use-case diagram, který zobrazuje funkčnosti systému. V diagramu jsou dva aktéři, Jedním je uživatel, který interaguje se hrou a druhým je systém. Systém reprezentuje software, který na pozadí řídí logiku hry. Diagram je zobrazený na následujícím obrázku.



Obrázek 9: Use-case diagram mobilní hry

4.4 Wireframy

Plán hry je definovaný, a tak je možnost přejít na přípravy wireframů, dle kterých bude hra následně vytvářena. Při spuštění aplikace by hráč neměl být přímo puštěn do samotné hry, a tak by bylo vhodné vytvořit nějaké úvodní menu hry, ve kterém uživatel hru odstartuje nebo ji vypne.



Obrázek 10: Wireframe pro úvodní obrazovku hry

První wireframe byl vytvořený pro úvodní obrazovku hry, ve wireframu je název hry a dvě tlačítka. První tlačítko načte druhý wireframe, který zobrazuje průběh hry. Druhé tlačítko je určené pro vypnutí aplikace.



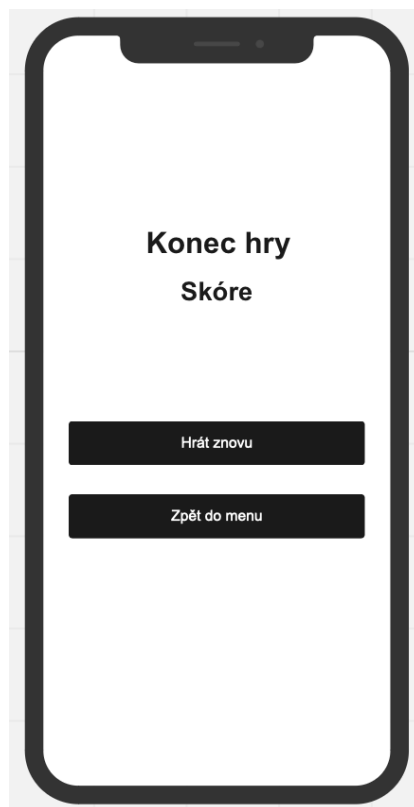
Obrázek 11: Wireframe pro nastavení hry

Druhý zobrazuje nastavení, ve kterém si uživatel vybírá, jaké početní operace chce, aby mu aplikace generovala. Ve druhém wireframu je úvodní text, čtyři políčka určená pro zaškrtnutí a tlačítko, které přesune hráče do hry.



Obrázek 12: Wireframe samotné hry

Další wireframe, který byl vytvořen, zobrazuje hlavní část hry. V té se hráč snaží vyřešit matematický příklad. Ve wireframu jsou čtyři tlačítka, každé z nich zobrazí možný výsledek příkladu. Po kliknutí na tlačítko se buďto odebere pokus nebo přičte bod do skóre. Příklad se zobrazuje přímo nad tlačítky s výsledky. Nad příkladem se objeví informace o tom, zda hráč zvolil správný nebo špatný výsledek. Pokusy se zobrazují v levém horním rohu wireframu, každý pokus je zobrazen v podobě srdce. Skóre hráče je zobrazeno naopak v pravém horním rohu obrazovky. Wireframe dále zobrazuje čas, během kterého musí hráč vyřešit matematický příklad, ten má podobu barevné čáry, která postupně mizí po dobu deseti vteřin.



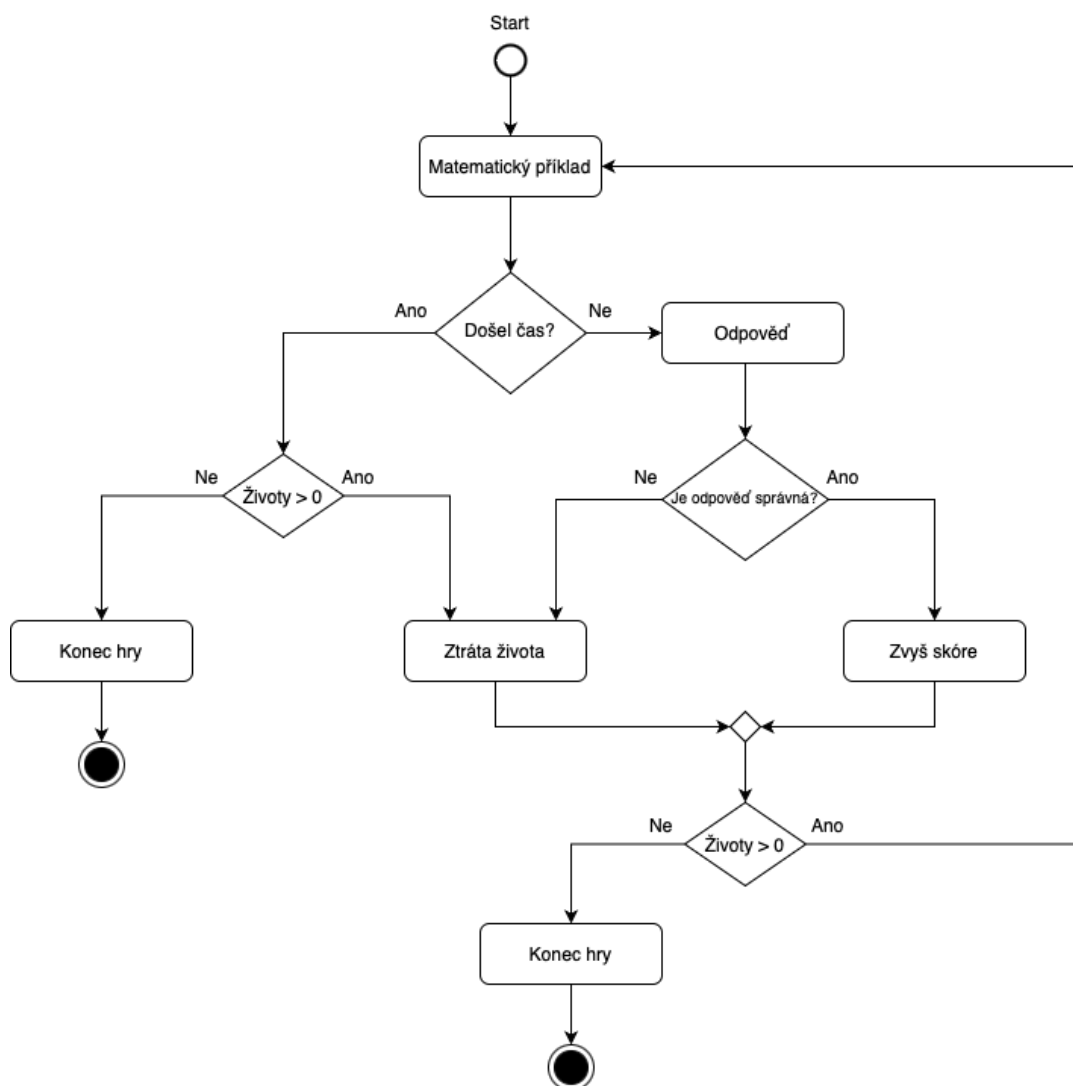
Obrázek 13: Wireframe pro konec hry

Poslední wireframe, byl vytvořen pro konec hry. Zobrazí se v tu chvíli, kdy uživateli dojdou všechny pokusy na to vyřešit příklad. Wireframe zobrazí text, který informuje hráče o tom, že jeho hra skončila a kolik získal bodů. Pod těmito informacemi se zobrazí dvě tlačítka. Pokud hráč klikne na první tlačítko celá hra začne znovu, pokud klikne na druhé zobrazí se první wireframe hry, hráč se vrátí zpět do menu celé hry.

4.5 Vývojový diagram

Podle požadavků pro hru je vhodné vytvořit také vývojový diagram, který by znázorňoval průběh hry. Následující diagram znázorňuje, jak hra funguje a co stane na základě toho jaký hráč zvolí výsledek matematického příkladu. Ze všeho nejdříve se zavolá funkce, která zobrazí matematický příklad. Pokud hráči dojde čas a životy, hra je u konce a funkce konec hry aktivuje obrazovku s výsledným skóre. Pokud hráč ještě nějaké životy má, tedy je splněna podmínka, i přesto mu došel čas a ztrácí život. Pokud není splněná podmínka o tom, zda hráči došel čas, vybírá odpověď neboli výsledek příkladu. Pokud je zvolen výsledek správně, hráč získává jeden bod. Pokud je zvolen špatný výsledek, hráč ztrácí život. Následně pokud není splněná podmínka a hráč už žádný život nemá, hra je u konce a opět

se aktivuje obrazovka s výsledným skóre. Pokud je splněná podmínka a hráč má více jak nula životů, znovu se spustí funkce na generování matematického příkladu.

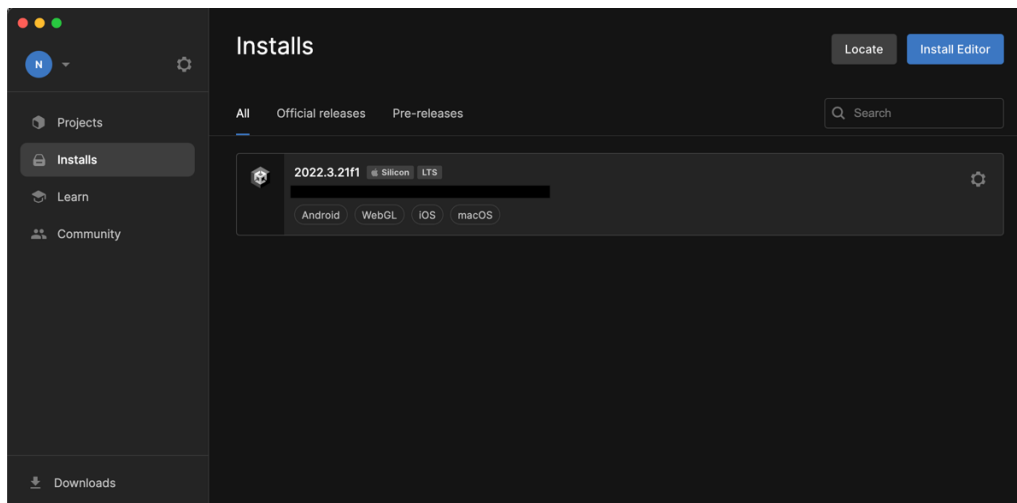


Obrázek 14: Vývojový diagram

4.6 Vhodný engine

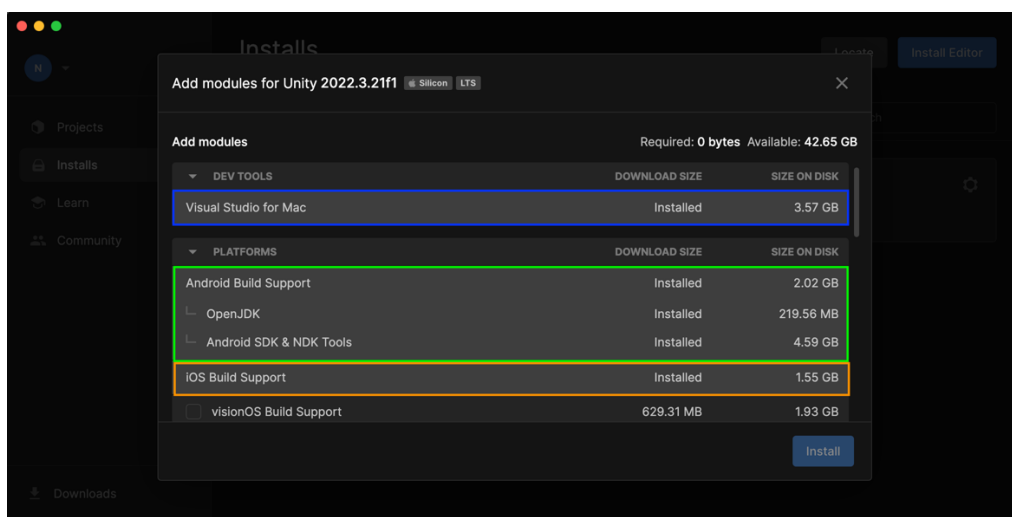
V teoretické části práce byly představeny vhodné nástroje, kterými lze vyvíjet mobilní hry. K vývoji této hry byl vybrán Unity engine. Vybrán byl z toho důvodu, protože je to jeden z nejvíce používaných nástrojů, co se týče vývoje her. Má velkou komunitu vývojářů. Poskytuje rozsáhlou a srozumitelnou dokumentaci. Podporuje vývoj her na více platform. Možnost vývoje 2D a 3D mobilních her. Je to vhodný nástroj pro začátečníky. Z těchto důvodů je to vhodný nástroj pro tvorbu této hry.

Nejdříve je potřeba instalace Unity, která je poměrně jednoduchá. Stažení programu je možné na stránce Unity.com. Po instalaci se spustí Unity Hub, který vyžaduje registraci a přihlášení uživatele.



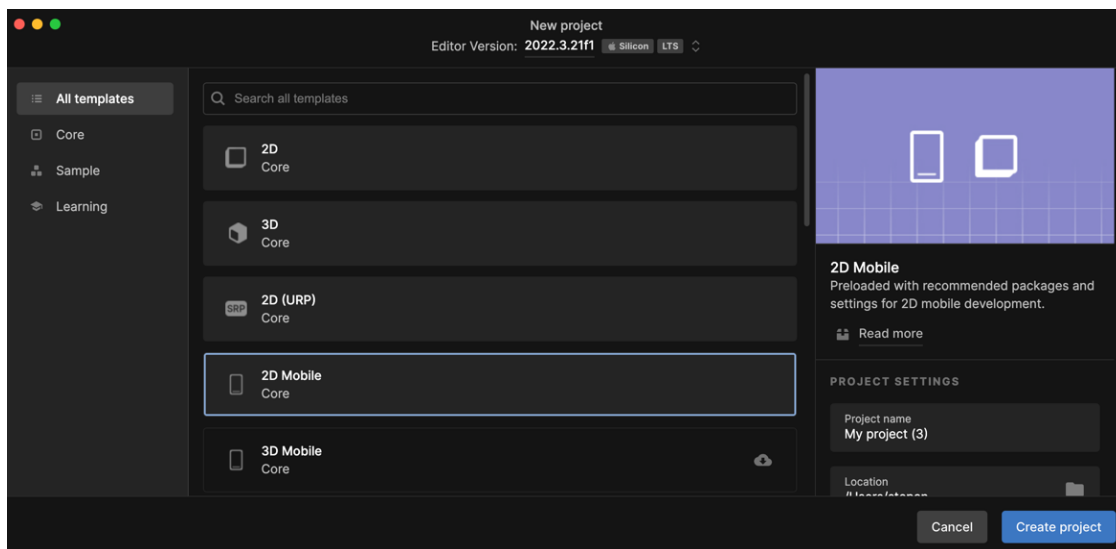
Obrázek 15: Unity hub

Jakmile je uživatel přihlášený zobrazí se okno s možností instalovat editor. Tento krok je nutný pro další používání program. Pro založení projektu je tedy nejdříve nutné nainstalovat Editor. To se provede pomocí modrého tlačítka v pravém horním rohu. Jakmile je instalace hotová, tak je potřeba také nainstalovat Visual Studio, které je důležité pro psaní skriptů. Dále je také potřeba nainstalovat moduly, které pomáhají k sestavení výsledného projektu jako aplikaci pro mobilní zařízení Android a iOS.



Obrázek 16: Instalace modulů

Následně je možné založit projekt a spustit editor. V záložce projekty je potřeba kliknout na nový projekt, poté vybrat šablonu projektu, vybrat místo pro uložení projektu a název projektu.

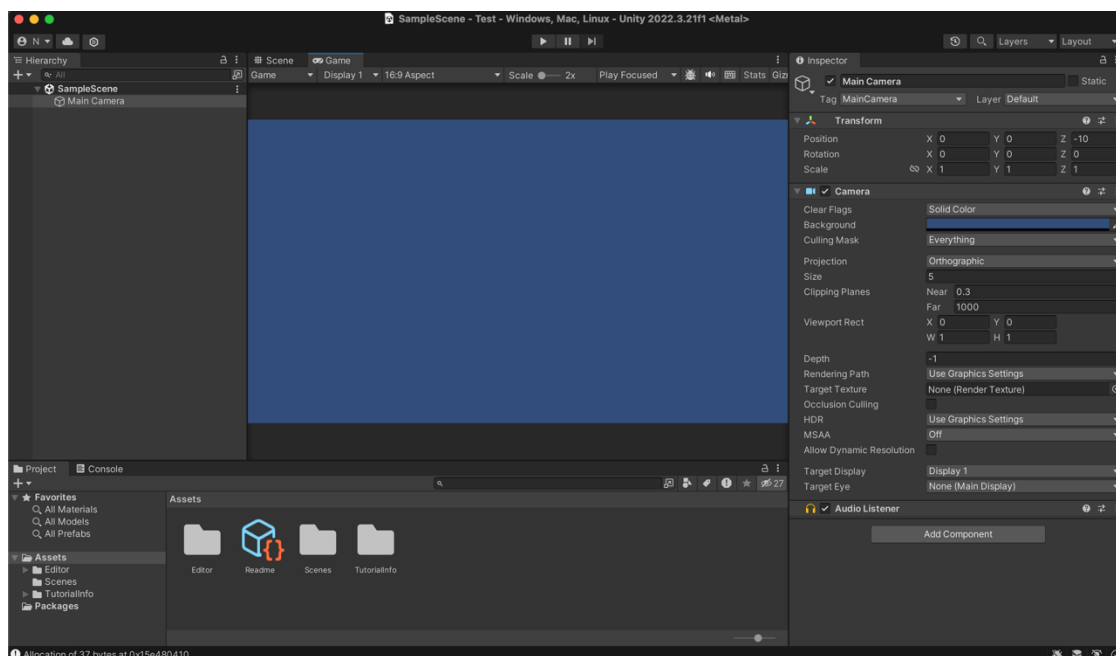


Obrázek 17: Šablony pro projekt

4.7 Vývoj hry

4.7.1 Příprava scén

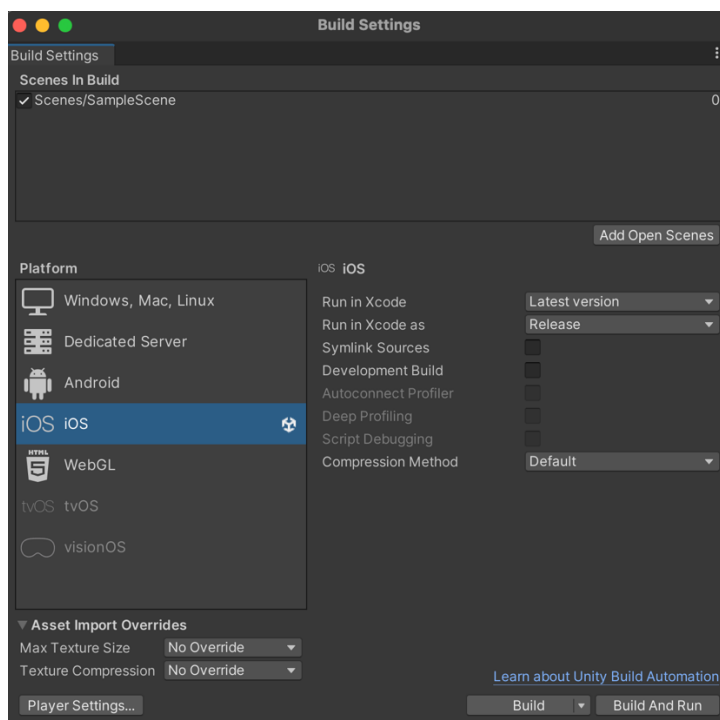
Jakmile je projekt vytvořený otevře se samotné Unity. Přímo uprostřed obrazovky je scéna se hrou, kde se zobrazují všechny prvky hry, jako text, tlačítka a objekty. Je to místo, kde se zobrazuje scéna hry a kde se upravuje. Vlevo je místo pro vytváření prvků hry. V této části rozhraní se vytvářejí veškeré texty, tlačítka a různé další prvky vhodné pro hru.



Obrázek 18: Rozhraní unity

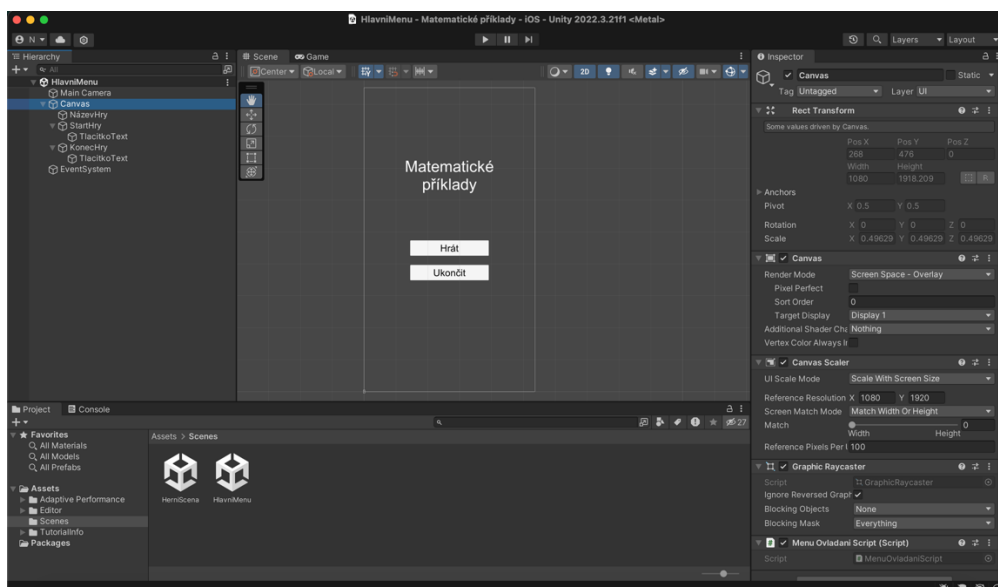
Dolní část rozhraní zobrazuje všechny assety hry, jako třeba obrázky a skripty. Pomocí dolní části je také možné přepínat scény celé hry. Pomocí části vpravo je možné s prvky hry různě pracovat, například přiřadit objektu obrázek.

Pro vytvoření požadované hry je potřeba nejdříve upravit nastavení scény. K tomu je potřeba přejít do nastavení projektu a přepnout cílovou platformu na jedno z mobilních zařízení.



Obrázek 19: Nastavení buildu

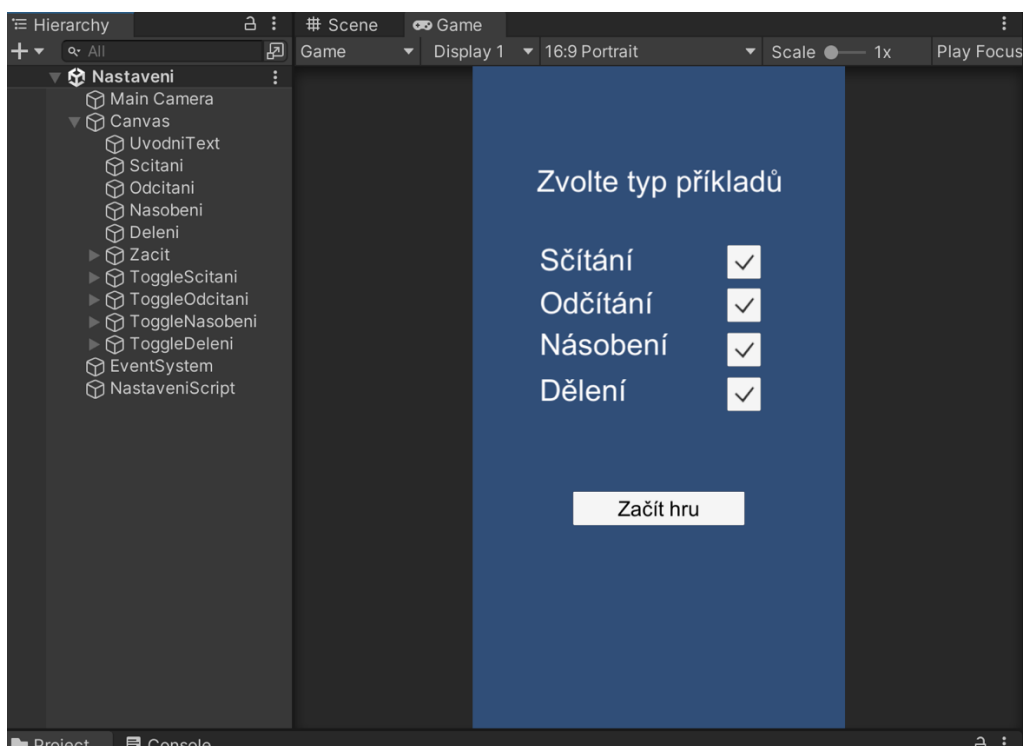
Následně byly vytvořeny dvě scény jedna pro hlavní menu hry a druhá pro samotnou hru, ve které se generují příklady a hráč vybírá výsledky. První scéna se jmenuje „HlavniMenu“ a byl do ní přidán textový objekt a dvě tlačítka s textem.



Obrázek 20: První scéna

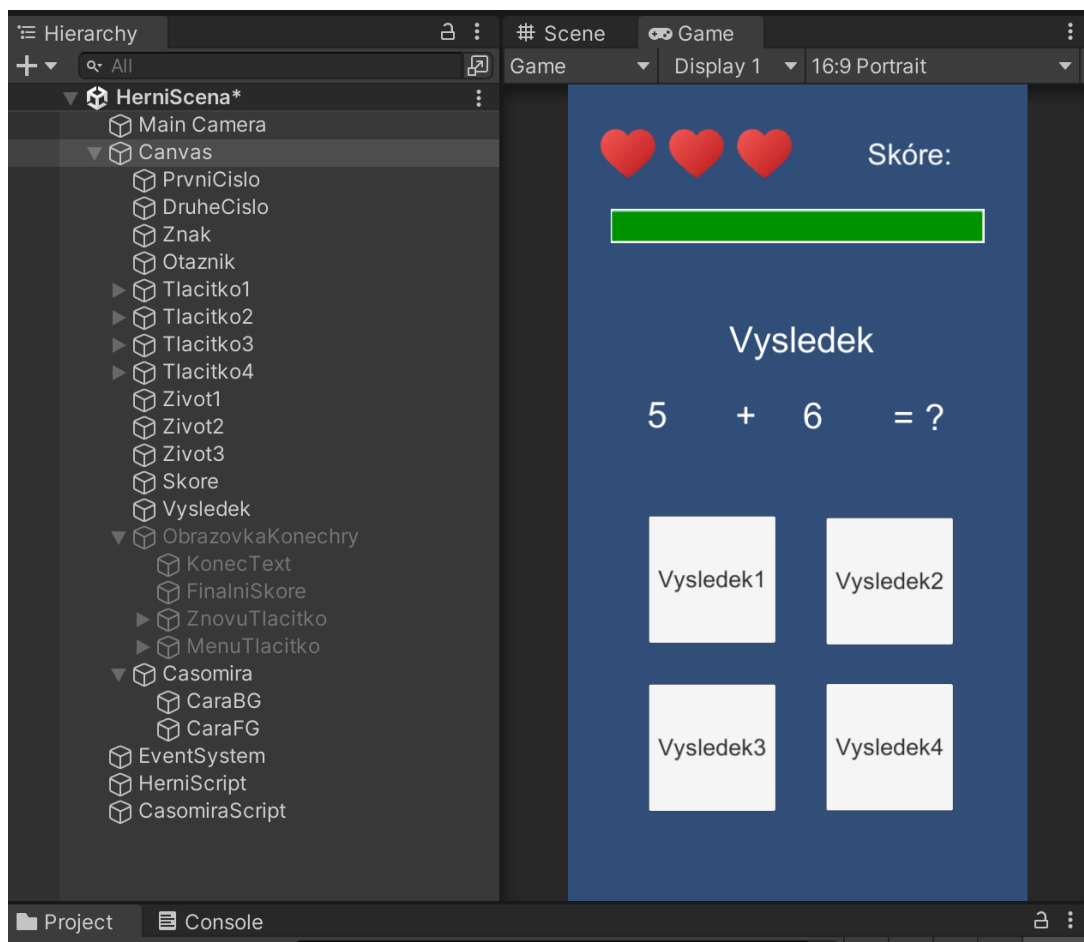
Textový objekt zobrazuje název hry. První tlačítko zobrazí druhou scénu, to pro hráče to znamená, že odstartuje hru. Druhé tlačítko vypne aplikaci. Scéně bylo dále upraveno rozlišení na 1080 x 1920. Dále byla upravena tak, aby se veškeré prvky zobrazovaly stejně bez ohledu na to, jak velká bude obrazovka.

Druhá scéna byla vytvořena pro nastavení hry, ve kterém hráč vybírá, jaké početní operace bude hra generovat. Scéna se skládá z několika textových objektů, jednoho tlačítka a čtyř zaškrťovacích políček.



Obrázek 21: Druhá scéna

V poslední scéně byly vytvořené textové prvky, pomocí kterých se zobrazuje matematický příklad. Ve scéně se také nachází tlačítka pro výsledky příkladu. Potom tři životy, které reprezentují pokusy na vyřešení příkladu. Dále textový prvek pro skóre, znak v příkladu a výsledek, který zobrazuje informaci o tom, zda hráč zvolil správný nebo špatný výsledek. Ve scéně se také nacházejí prvky pro čas a poslední obrazovku hry.



Obrázek 22: Hlavní scéna hry

4.7.2 Skripty

Pro úvodní scénu byl vytvořený skript a ten se následně přiřadil k plátnu v rozhraní unity. Skript obsahuje dvě metody.

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class MenuOvladaniScript : MonoBehaviour
{
    // Start is called before the first frame update
    public void StartHry()
    {
        SceneManager.LoadSceneAsync("Nastaveni");
    }

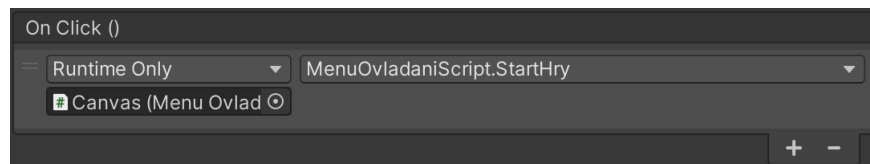
    public void KonecHry()
    {
        Application.Quit();
    }

    // Update is called once per frame
    void Update()
    {
    }
}

```

Kód 1: Skript pro úvodní obrazovku

První metoda říká, aby se načetla scéna s názvem „Nastavení“ a druhá vypne aplikaci. První metoda byla přiřazena tlačítku StartHry a druhá tlačítku KonecHry. Do skriptu byl také zapsán příkaz „using UnityEngine.SceneManagement“ aby hra měla možnost pracovat se scénami.



Obrázek 23: Přiřazení skriptu pro tlačítko start hry

Následný skript ukládá hráčovo nastavení pro zobrazování příkladů. Na začátku jsou definované proměnné pro zaškrťovací pole. Metoda „UlozaHraj“ nastaví zaškrťovací pole, uloží toto nastavení a načte hlavní scénu hry. Tato metoda byla přiřazena na tlačítko ve scéně projektu.

```

public class NastaveniScript : MonoBehaviour
{
    public Toggle tScitani;
    public Toggle tOdcitani;
    public Toggle tNasobeni;
    public Toggle tDeleni;

    public void UlozaHraj()
    {
        PlayerPrefs.SetInt("VybranoScitani", tScitani.isOn ? 1 : 0);
        PlayerPrefs.SetInt("VybranoOdcitani", tOdcitani.isOn ? 1 : 0);
        PlayerPrefs.SetInt("VybranoNasobeni", tNasobeni.isOn ? 1 : 0);
        PlayerPrefs.SetInt("VybranoDeleni", tDeleni.isOn ? 1 : 0);
        PlayerPrefs.Save();
        SceneManager.LoadSceneAsync("HerniScena");
    }
}

```

Kód 2: Ukládání nastavení

Celá hlavní scéna hry funguje pomocí skriptu „HerniScript“, ve kterém jsou metody pro zobrazení příkladu, fungování tlačítek, přičítání bodů a snižování životů. Nejdříve je třeba deklarovat proměnné pro metody a textové objekty.

```

public class HerniScript : MonoBehaviour
{
    public Text PrvniCislo;
    public Text DruheCislo;
    public Text Znak;
    public Text Odpoved1;
    public Text Odpoved2;
    public Text Odpoved3;
    public Text Odpoved4;

    public int nahodnePrvniCislo;
    public int nahodneDruheCislo;

    int PrvniCisloVPrikladu;
    int DruheCisloVPrikladu;
    int SpravnaOdpoved;
    int[] Odpovedi = new int[4];
    int TypOperace;
    public int MistoSpravneOdpovedi;

    public Text Vysledek;

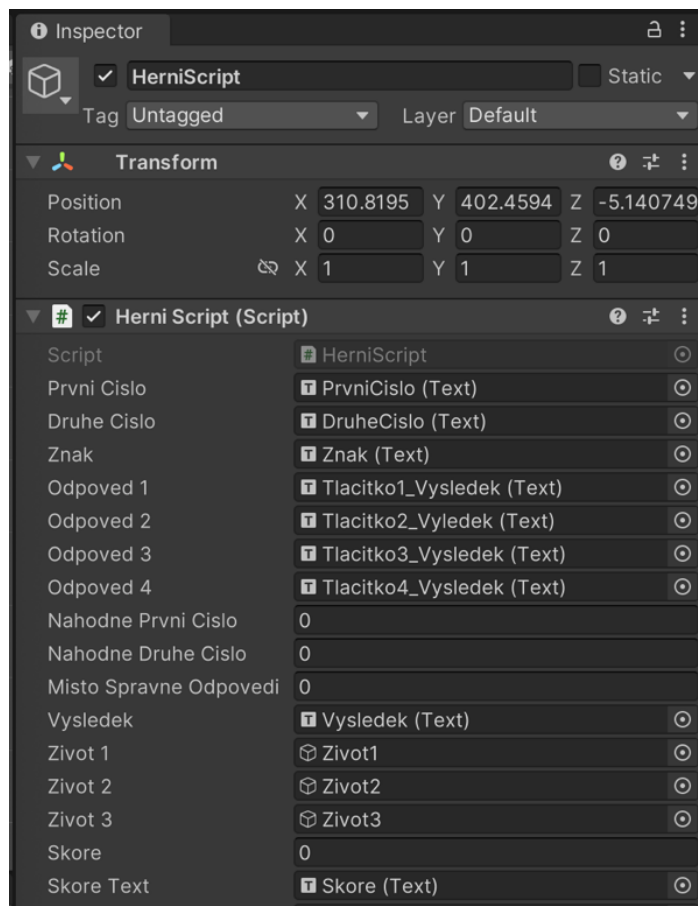
    public GameObject Zivot1;
    public GameObject Zivot2;
    public GameObject Zivot3;
    private int zivoty = 3;

    public int skore = 0;
    public Text SkoreText;
}

```

Kód 3: Definice proměnných

Většina z těchto proměnných jsou nastaveny jako veřejné, a proto jsou viditelné v inspektoru Unity. Díky tomu je možné k proměnným přiřadit objekty, které byly dříve vytvořeny jako text, tlačítka a životy.



Obrázek 24: Přiřazení objektů k proměnným

Pro zobrazení matematického příkladu ve hře byla vytvořena funkce „MatematickyPriklad“. Začátek metody vytváří seznam, který obsahuje číselné identifikátory matematických operací, jež jsou povoleny pro generování příkladů. Potom načítá nastavení, které provedl uživatel. Pokud by se stalo, že by uživatel nevybral žádnou operaci, automaticky se do seznamu přidá identifikátor nula pro sčítání. Tato část skriptu zajišťuje, že alespoň jedna operace bude vždy k dispozici pro generování příkladů.

```

public void MatematickyPriklad()
{
    List<int> povoleneOperace = new List<int>();
    if (PlayerPrefs.GetInt("VybranoScitani", 0) == 1) povoleneOperace.Add(0);
    if (PlayerPrefs.GetInt("VybranoOdcitani", 0) == 1) povoleneOperace.Add(1);
    if (PlayerPrefs.GetInt("VybranoNasobeni", 0) == 1) povoleneOperace.Add(2);
    if (PlayerPrefs.GetInt("VybranoDeleni", 0) == 1) povoleneOperace.Add(3);

    if (povoleneOperace.Count == 0)
    {
        povoleneOperace.Add(0);
    }

    nahodnePrvniCislo = UnityEngine.Random.Range(1, 10);
    nahodneDruheCislo = UnityEngine.Random.Range(1, 10);
    PrvniCisloVPrikladu = nahodnePrvniCislo;
    DruheCisloVPrikladu = nahodneDruheCislo;

    TypOperace = povoleneOperace[UnityEngine.Random.Range(0, povoleneOperace.Count)];

    switch (TypOperace)
    {

```

Kód 4: Funkce pro generování matematického příkladu

Skript dále vygeneruje dvě náhodná čísla, které potom přiřadí do dalších proměnných. Následně přiřazuje náhodné číslo do proměnné „TypOperace“ ze seznamu povolených operací. Tento náhodný výběr určuje, který typ matematické operace bude použit pro generování aktuálního příkladu. Podle toho, jaké číslo se vybere bude mít proměnná „SpravnaOdpoved“ informaci o tom jaký typ příkladu bude probíhat.

```

switch (TypOperace)
{
    case 0:
        SpravnaOdpoved = PrvniCisloVPrikladu + DruheCisloVPrikladu;
        Znak.text = "+";
        break;
    case 1:
        if (nahodnePrvniCislo < nahodneDruheCislo)
        {
            int a = nahodnePrvniCislo;
            nahodnePrvniCislo = nahodneDruheCislo;
            nahodneDruheCislo = a;
        }
        PrvniCisloVPrikladu = nahodnePrvniCislo;
        DruheCisloVPrikladu = nahodneDruheCislo;
        SpravnaOdpoved = PrvniCisloVPrikladu - DruheCisloVPrikladu;
        Znak.text = "-";
        break;
    case 2:
        SpravnaOdpoved = PrvniCisloVPrikladu * DruheCisloVPrikladu;
        Znak.text = "*";
        break;
    case 3:
        SpravnaOdpoved = PrvniCisloVPrikladu / DruheCisloVPrikladu;
        Znak.text = "/";
        PrvniCisloVPrikladu = SpravnaOdpoved * DruheCisloVPrikladu;
        break;
}

```

Kód 5: Genrování příkladů

Dále je do proměnné „MistoSpravneOdpovedi“ přiřazeno číslo, které reprezentuje kde se nachází správná odpověď. Funkce dále nastaví pole „Odpovedi“ na nulu, což vyčistí pole, aby se do něj mohli vložit nové. Následně na to probíhá generování falešných odpovědí pro ostatní tlačítka, tím způsobem, že pro každou pozici v poli „Odpovedi“, která není pozicí správné odpovědi, se vygeneruje náhodná odpověď blízka správné odpovědi (s rozdílem od -3 do 3), ale tak, aby nebyla stejná jako správná odpověď nebo jakákoli jiná již existující odpověď v poli. Metoda následně přiřadí odpovědi do textových polí.

```

MistoSpravneOdpovedi = UnityEngine.Random.Range(0, 4);

for (int i = 0; i < Odpovedi.Length; i++)
{
    Odpovedi[i] = 0;
}

Odpovedi[MistoSpravneOdpovedi] = SpravnaOdpoved;

for (int i = 0; i < Odpovedi.Length; i++)
{
    if (i != MistoSpravneOdpovedi)
    {
        int nahodnaOdpoved;
        do
        {
            nahodnaOdpoved = SpravnaOdpoved + UnityEngine.Random.Range(-3, 4);
        } while (nahodnaOdpoved == SpravnaOdpoved || Array.Exists(Odpovedi, element => element == nahodnaOdpoved));
        Odpovedi[i] = nahodnaOdpoved;
    }
}

Odpoved1.text = "" + Odpovedi[0];
Odpoved2.text = "" + Odpovedi[1];
Odpoved3.text = "" + Odpovedi[2];
Odpoved4.text = "" + Odpovedi[3];

PrvniCislo.text = "" + PrvniCisloVPrikladu;
DruheCislo.text = "" + DruheCisloVPrikladu;
}

```

Kód 6: Pokračování funkce pro generování příkladů

Pro každé tlačítko byla vytvořena metoda, která se zavolá v případě kdy uživatel klikne na tlačítko. Metody jsou vždy stejné pro všechny čtyři tlačítka, rozdílné jsou pouze v názvu.

```

public void TlacitkoOdpoved1()
{
    if(MistoSpravneOdpovedi == 0)
    {
        Vysledek.enabled = true;
        Vysledek.color = Color.green;
        Vysledek.text = ("Správně!");
        ZvysSkore();
        casomiraScript.ResetujCas();
        Invoke("Smazat", 1);
    }
    else
    {
        Vysledek.enabled = true;
        Vysledek.color = Color.red;
        Vysledek.text = ("Špatně, zkus to znovu");
        ZtrataZivota();
        casomiraScript.ResetujCas();
        Invoke("Smazat", 1);
    }
}

```

Kód 7: Funkce pro tlačítka s výsledky

Metoda „TlacitkoOdpovedi1“ říká, že pokud se dané tlačítko zvolí a místo správné odpovědi se rovná nule, pak se proměnná s výsledkem zviditelní, zbarví se na zelenou barvu a vygeneruje text „Správně!“. Následně na to se provedou další tři metody. První metoda zvýší skóre, druhá resetuje časomíru a třetí po jedné vteřině smaže matematický příklad. Pokud je zvoleno toto tlačítko, ale podmínka není splněna, pak se znovu z aktivní proměnná s tím rozdílem, že nebude mít zelenou barvu, ale červenou a vygeneruje text „Špatně, zkus to znovu“. Potom znovu provede další tři metody, ale jedna metoda je v tomto případě rozdílná. V tomto případě se neprovede metoda pro zvýšení skóre, ale naopak pro ztrátu života. Zmíněné čtyři metody jsou zobrazeny v následujícím obrázku.

```
public void Smazat()
{
    if(Vysledek != null)
        Vysledek.enabled = false;
    MatematickyPriklad();
}

public void ZtrataZivota()
{
    zivoty -= 1;
    switch (zivoty)
    {
        case 2:
            Zivot3.SetActive(false);
            break;
        case 1:
            Zivot2.SetActive(false);
            break;
        case 0:
            Zivot1.SetActive(false);
            break;
    }
}

public void ZvysSkore()
{
    skore += 1;
    SkoreText.text = "Skóre: " + skore;
}
```

Kód 8: Funkce pro smazání příkladu, ztrátu životů a zvýšení skóre

Metoda s názvem „Smazat“ skryje UI element „Vysledek“ pokud je tento element nulový a znovu zavolá metodu „MatematickyPriklad“, která vygeneruje nový matematický příklad.

Metoda „ZtrataZivota“ řídí, co se stane, když hráč odpoví špatně. Nejdříve sníží počet životů o jeden a potom vzhledem na aktuálním počtu zbývajících životů deaktivuje odpovídající element reprezentující životy.

Metoda „ZvysSkore“ se stará o zvýšení hráčova skóre po správné odpovědi. Nejdříve přičte jedničku k hráčovu skóre a potom aktualizuje „SkoreText“ tak, aby zobrazil novou hodnotu skóre. Celkově jsou tyto metody navrženy tak, aby poskytovaly zpětnou vazbu hráči a řídily herní stav v závislosti na tom, zda hráč odpověděl správně nebo ne.

Aby fungovaly prvky vytvořené pro čas potřebný k vyřešení příkladu, byl vytvořen skript „CasomiraScript“. Ten začíná definicí proměnných. První proměnná reprezentuje vizuální zobrazení časoměry. Druhá privátní proměnná slouží pro sledování zbývajících času na časoměře. Třetí proměnná je veřejná a pomáhá nastavit maximální čas pro vyřešení matematického příkladu. Poslední proměnná je také veřejná a odkazuje na hlavní skript hry. Díky této proměnné umožňuje daný skript komunikovat s hlavním skriptem hry. Metoda „Start“ se volá automaticky při začátku hry a nastaví proměnnou „ZbyvajiciCas“ na hodnotu „MaximalniCas“, což znamená, že časomíra začíná s plným časem při startu nebo restartu hry. Metoda „Update“ je automaticky volána jednou za každý snímek pro každý skript, ve kterém je definována. Pokud je nějaký „ZbyvajiciCas“, odčítá se od něj „Time.deltaTime“, což je doba, která uplynula od posledního snímku, díky čemuž se simuluje odpočítávání. Další řádek slouží k dynamickému upravování obrázku na základě poměru mezi zbývajícím časem a maximálním časem, který byl nastaven pro časomíru. Poslední metody skriptu nastaví zbývající čas zpět na maximální čas, díky čemuž resetuje časomíru.

```

public class CasomiraScript : MonoBehaviour
{
    public Image CaraCas;
    float ZbyvajiciCas;
    public float MaximalniCas = 11.0f;
    public HerniScript herniScript;

    // Start is called before the first frame update
    void Start()
    {
        ZbyvajiciCas = MaximalniCas;
    }

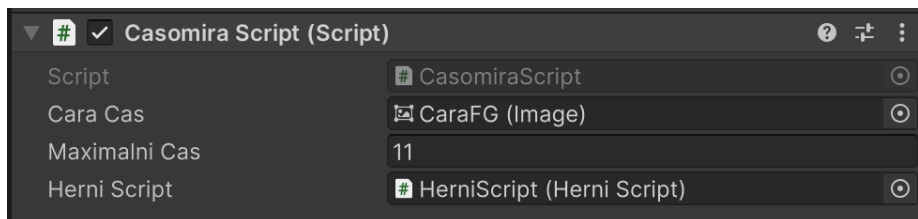
    // Update is called once per frame
    void Update()
    {
        if (ZbyvajiciCas > 0)
        {
            ZbyvajiciCas -= Time.deltaTime;
            CaraCas.fillAmount = ZbyvajiciCas / MaximalniCas;
        }
        else
        {
            herniScript.ZtrataZivota();
            ResetujCas();
        }
    }

    public void ResetujCas()
    {
        ZbyvajiciCas = MaximalniCas;
    }
}

```

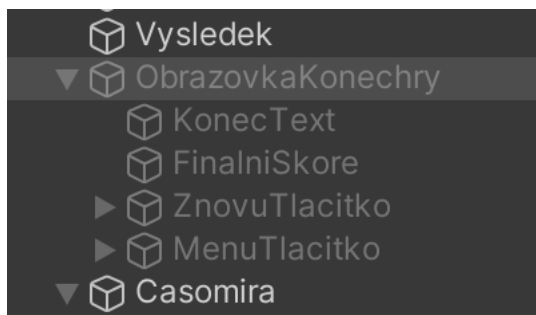
Kód 9: Skript pro řízení časomíry

V inspektoru Unity byly následně přiřazeny všechny objekty k daným proměnným.



Obrázek 25: Přiřazení proměnných

Obrazovka, která se zobrazí potom co hráč dohraje, se vytvořila pomocí panelu. Ten není na scéně hry vidět, protože je během hry neaktivní. Že je obrazovka neaktivní také naznačuje zašedlá barva textu všech prvků, včetně obrazovky, přiřazených k panelu.



Obrázek 26: Objekty obrazovky pro konec hry

Kód pro tento panel byl vložen do již vytvořeného hlavního skriptu hry. V něm byly vytvořeny proměnné pro objekt a text, který bude zobrazovat finální skóre. Do tohoto skriptu byly také přidány proměnné pro časomíru.

```
public GameObject ObrazovkaKonecHry;
public Text FinalniSkore;

public CasomiraScript casomiraScript;
public GameObject Casomira;
```

Kód 10: Definování proměnných pro časomíru

Nejdříve bylo třeba určit co by se mělo stát, pokud by hra měla skončit. To bylo nastaveno v metodě „KonecHry“. Metoda deaktivuje časomíru, aby nebyla vidět. Zobrazí panel, který funguje jako obrazovka pro konec hry. K tomu také zobrazí finální skóre, kterého hráč dosáhl. Zároveň byly vytvořeny funkce pro dvě tlačítka. První restartuje hru tak, že znovu načte danou scénu. Druhé vrátí hráče do hlavního menu hry, tedy načte první scénu v projektu.

```
public void KonecHry()
{
    Casomira.SetActive(false);
    ObrazovkaKonecHry.SetActive(true);
    FinalniSkore.text = "Skóre: " + skore;
}

public void RestartHry()
{
    SceneManager.LoadScene(SceneManager.GetActiveScene().name);
}

public void ZpetDoMenu()
{
    SceneManager.LoadScene("HlavniMenu");
}
```

Kód 11: Konec hry a funkce pro tlačítka

Do metody, která určuje, co se provede po startu hry byla importována již vytvořená metoda pro zobrazení matematického příkladu. Konec hry nastane ve chvíli, kdy hráči nezbydou žádné životy, což je kontrolováno pomocí update.

```
void Start()
{
    MatematickyPriklad();
}

void Update()
{
    if (zivoty <= 0)
    {
        KonecHry();
    }
}
```

Kód 12: Start hry a kontrola životů

4.8 Grafika hry

Ve hře bylo několik prvků, které postrádaly grafickou stránku. Proto byla hra následně vylepšena, aby byla na pohled zajímavější a poutavější. Díky tomu může hra přispívat k lepšímu uživatelskému zážitku. Atraktivní vizuální stránka může zároveň zvýšit šance hry na úspěch v konkurenčním prostředí.

Nejvíce viditelné prvky hry jsou samotné texty, na které uživatel naráží na každé scéně hry. Proto byl každému textu přiřazený styl písma Bad Comic.

Bad Comic

Obrázek 27: Styl písma

Jedná se o styl písma, který na první pohled vypadá hravě a neseriózně. Vzhledem k tomu, že se jedná o hru pro děti tak je to vhodný styl písma pro text této hry. Styl písma byl importován do Unity a následně přiřazen na každý prvek textu. V některých částech hry, jako úvodní menu a nastavení, byl text také zvýrazněn černým obtažením textu a stínováním.

Matematické příklady

Obrázek 28: Příklad zobrazení stylu písma

Pro zlepšení vizuální stránky hry byly také upraveny všechny tlačítka ve hře. Design pro všechna tlačítka, které přepínají scény hry byla vytvořena naprosto stejným způsobem. Rozdílné jsou pouze jejich barvy a text. Všechny grafické prvky hry byly vytvořeny v programu Adobe Photoshop.



Obrázek 29: Design tlačítek

Úpravou prošlo také pozadí celé hry. Byla změněna barva pozadí z modré na světlemodrou. Do pozadí hry se následně přidala další vrstva. Tato vrstva se skládá z jednotlivých znamének, které se používají v matematických příkladech hry. Aby vrstva nijak nezasahovala do ostatních prvků a nekazila zážitek ze hry, byla jí nastavena průhlednost. Hodnota průhlednosti byla na prvních dvou scénách hry nastavena stejně, ale pro samotnou scénu hry byla hodnota ještě zvýšena, aby generované příklady byly více vidět a uživatel se mohl lépe soustředit na jejich řešení.



Obrázek 30: Pozadí hry

V hlavní scéně hry byl také upraven design jednotlivých tlačítek, které zobrazují výsledky matematických příkladů. Tlačítka byla pouze zvýrazněna pomocí stínování, aby se jevila jako skutečná a zároveň tvořila vhodné pozadí pro zobrazený výsledek. Na závěr byla pro celou hru vytvořena také ikonka, která je viditelná po nainstalování hry do mobilního zařízení. Ikonka, stejně jako pozadí hry, zobrazuje jednotlivé znaky matematických příkladů, tedy znak pro sčítání, odčítání, násobení a dělení.

4.9 Testování na mobilním zařízení

Pro otestování aplikace je nutné postavit build projektu. K tomu je potřeba vybrat cílový operační systém, na kterém bude hra testována. Nejdříve byl vybrán jako cílový operační systém iOS a poté Android. K testování aplikace byla použita následující zařízení.

ID	Mobilní zařízení	Operační systém
1	iPhone 13	iOS 17.4.1
2	iPhone 5s	iOS 12.5.5
3	Samsung Galaxy S8	Android 9
4	Huawei P20 Lite	Android 9
5	Samsung Galaxy S22	Android 14

Tabulka 2: Zařízení určené k testování

Pro otestování aplikace na zařízení s operačním systémem iOS byla, skrze Unity, vytvořena složka se soubory aplikace. Aplikaci je možné otestovat pouze pomocí počítače s operačním systémem MacOS. K testování je také nutné mít nainstalovaný program Xcode.

V aplikaci Xcode je za potřebí stažení souborů, které podporují testování pomocí iOS zařízení. Na testovaná zařízení byla postupně instalována matematická hra. Žádné iOS zařízení nevykazovalo problémy při instalaci aplikace. Během testování byla aplikace stabilní a nikdy se sama nevyhnula. Celý proces testování na těchto zařízeních proběhl v pořádku, přičemž byly otestovány veškeré funkce aplikace.



Obrázek 31: Testování na zařízení iOS

Pro otestování aplikace na zařízeních s operačním systémem Android byl, skrze Unity, vygenerován jediný soubor apk. Tento soubor se následně nahrál na všechna zařízení určená k testování aplikace. Instalace aplikace na těchto zařízeních proběhla v pořádku. To samé platilo i během testování. Celý proces testování byl tedy dokončen i pro zařízení s operačním systémem Android.

Proces testování aplikace na Android zařízení je v porovnání s testováním na iOS mnohem jednodušší. Pro Android zařízení je nutný pouze jediný soubor, který je možné pomocí počítače nahrát do zařízení a potom aplikaci nainstalovat. Zatímco u testování aplikace na iOS zařízení je zapotřebí počítač s MacOS systémem a také program Xcode. Celý proces je proto složitější a mnohem časově náročnější.

Aplikace na všech zařízeních fungovala správně a přesně tak jak byla vytvořena. Avšak byl nalezen problém během převrácení obrazovky do vodorovné polohy. Při převrácení obrazovky dochází k rozhození rozlišení aplikace. K tomuto aplikaci nebyla přizpůsobena. Tento problém může být předmětem další aktualizace.

Dále byla pozorována zvláštní změna barev v pozadí mezi všemi zařízeními. Třetí zařízení vykazovalo úplně jiný odstín modré barvy. Nejlépe barvu zobrazovala zařízení s operačním systémem iOS. Ostatní zařízení s operačním systémem Android zobrazovala barvu pozadí relativně přesně.

Aplikaci také otestovali dva uživatelé. První uživatel testoval aplikaci již během vytváření hry, kdy hra neměla vylepšený design. To uživatel považoval za zlepšení a vhodnou úpravu aplikace. Oba uživatelé během testování nenašli žádnou chybu v chodu aplikace. Z těchto důvodů lze označit testování za úspěšné, při kterém byly nalezeny náměty na další úpravu aplikace. Aplikace je v tomto momentě připravená k nahrání do obchodů obou zařízení.

5 Výsledky a diskuse

Cílem této práce byla tvorba edukativní matematické hry pro mobilní zařízení. Cílem literární části práce bylo popsání her, představit jejich vývoj od vzniku první mobilní hry a popsat přínosy pro společnost. Literární část se dále zabývala popisem vhodných enginů pro tvorbu her a procesem vývoje mobilních her. Vzhledem k tomu lze považovat cíle práce za splněné.

Při testování hry byly objeveny problémy se změnou rozlišení a barvou pozadí u jednoho testovacího zařízení. Těmto problémům by se dalo zamezit při budoucím vývoji hry. Testování i tak ukázalo, že aplikace funguje tak jak byla vytvořena a nyní je připravena pro nahrání do obchodů.

Přesto má aplikace několik nedostatků. Design hry mohl být mnohem propracovanější, aby více přitahoval potenciální uživatele. Hra mohla být také obohacena o nějaké zvuky. Například když uživatel zmáčkne tlačítko nebo když zvolí správný výsledek. Zároveň by také mohla mít animace, které by hře dodaly živější prezentaci. Všechny tyto prvky znamenají bohatší a poutavější zážitek ze hry, díky čemuž si uživatel může hru oblíbit. Zároveň by to mohlo pomoci navýšit šance na úspěch v konkurenčním prostředí a zlepšit první dojem ze hry.

6 Závěr

Diplomová práce se zabývá mobilními hrami a jejich vývojem. V literární části byla představena definice mobilních her, jejich historický vývoj, metody monetizace, různé kategorie, žánry a zároveň přínosy her pro společnost. Obsahem literární části je také popis herních enginů vhodných pro vývoj mobilních her. Dále je zde popsán proces vývoje mobilních her, který zahrnuje části jako tvorba designu, přípravy wireframů, use-case diagramů a vývojových diagramů.

Praktická část je zaměřena na vývoj mobilní matematické hry a detailní popis její tvorby od wireframů přes use-case a vývojový diagram až po popis skriptů hry. Celé hře byl upraven její design. Prvkům pro text byl přiřazen vhodný font a tlačítka hry byla graficky upravena, tak aby byla zajímavější. Hra byla následně otestovaná na mobilních platformách iOS a Android, což umožnilo ověřit její funkčnost.

Z této práce tedy vyplývá, že mobilní hry představují nejen významný segment herního průmyslu, ale také nabízejí jedinečné příležitosti pro vzdělání, rozvoj dovedností a společenský přínos. Praktická realizace matematické hry ukázala, jak teoretické postupy mohou být úspěšně aplikovány v praxi.

7 Seznam použitých zdrojů

- (1) OLEARY, Hana. What Are Mobile Games? *Cellularnews.com* [online]. 2023 [cit. 2024-01-26]. Dostupné z: <https://cellularnews.com/definitions/what-are-mobile-games/>
- (2) KARTHIKEYAN, Kaavya. The History, Evolution, and Future of Mobile Gaming. *Gameopedia* [online]. 2022 [cit. 2024-01-28]. Dostupné z: <https://www.gameopedia.com/the-history-evolution-and-future-of-mobile-gaming/>
- (3) From Tetris to Candy Crush: The History of Mobile Gaming. *Medium* [online]. 2020 [cit. 2024-01-28]. Dostupné z: <https://medium.com/@pubnative/from-tetris-to-candy-crush-the-history-of-mobile-gaming-da14436cf23b>
- (4) Tvůrce mobilní hry Clash of Clans loni prudce zvýšil zisk na 23 miliard. *E15* [online]. 2016 [cit. 2024-01-28]. Dostupné z: <https://www.e15.cz/magazin/tvurce-mobilni-hry-clash-of-clans-loni-prudce-zvysil-zisk-na-23-miliard-1278238>
- (5) VYHNANOVSKÝ, Ondřej. Pokémon Go věští budoucnost. Přichází svět rozšířené reality. *Lidovky.cz* [online]. 2024 [cit. 2024-02-09]. Dostupné z: https://www.lidovky.cz/ceska-pozice/pokemon-go-vesti-budoucnost-prichazi-svet-rozsirene-reality.A160819_104355_pozice-tema_houd
- (6) Základy monetizace mobilních aplikací. In: *Gamesdev* [online]. [cit. 2024-02-09]. Dostupné z: <https://gamesdev.cz/monetizace-mobilnich-aplikaci/>
- (7) Exploring the World of Mobile Game Genres. In: *Develop and publish game* [online]. [cit. 2024-02-09]. Dostupné z: <https://denebgames.com/blog/exploring-the-world-of-mobile-game-genres/>
- (8) MOTTA, Micah. Hybrid casual games: The new kid on the gaming app block. In: *Adjust* [online]. [cit. 2024-02-09]. Dostupné z: <https://www.adjust.com/blog/hybrid-casual-gaming-app-the-new-kid-on-the-block/>
- (9) World of Gaming — Game Categories, Genres, and Sub-Genres. In: *Medium* [online]. [cit. 2024-02-09]. Dostupné z: <https://medium.com/@99Games/world-of-gaming-game-categories-genres-and-sub-genres-c52f281a9fea>
- (10) Mobile game types 101: Know your gaming genres. In: *Adjust* [online]. [cit. 2024-02-09]. Dostupné z: <https://www.adjust.com/blog/mobile-game-types-for-app-developers/>
- (11) Understanding mobile games' genres. In: *Apptamin* [online]. [cit. 2024-02-09]. Dostupné z: <https://www.apptamin.com/blog/understanding-mobile-games-genres/>
- (12) 5 Positive Effects of Mobile Gaming You Probably Didn't Know. In: *The CSR Journal* [online]. [cit. 2024-02-09]. Dostupné z: <https://theCSRjournal.in/5-positive-effects-of-mobile-gaming-you-probably-didnt-know/>
- (13) GOSSETT, Stephen, POWERS, Jessica, ed. The Real Social Benefits of Video Games. In: *Builtin* [online]. [cit. 2024-02-09]. Dostupné z: <https://builtin.com/media-gaming/online-gaming-social-benefits>
- (14) Top 20 Educational Game Apps for Kids & Adults in 2024. *Capermint* [online]. 2014 [cit. 2024-03-31]. Dostupné z: <https://www.capermint.com/blog/educational-mobile-games/?fbclid=IwAR1IVSjB5tHsKtK08xppUiAEQ3eo2mNi7V5fJZWSvL31I4oE>

- R22V1b9CgFs_aem_AVcwGDaMh30o6sjrFzdtyRO8jltOK5c_3MFHs9H6hQb83r6yjIhKZOrU1qjeJ5lsHpW-n5tI76nBZN3FazdoxSl
- (15) Why your children really should play videogames. *Iberdrola* [online]. 2024 [cit. 2024-03-31]. Dostupné z: https://www.iberdrola.com/talent/benefits-video-games-learning?fbclid=IwAR2nLcoANw3S_J0fpZZsdQlnvU11CyqQafWJd1hYRjVagwQYKdLQHbGpGqk_aem_AVcI0CLMVbi6y4cmPygMsFYNi7tXpylyGrh5FXukVZWH-kB36u6hUaldG_unb8u4StBISuBt68Bnpj2Pl-SgDxKL
- (16) ŽOLTÁ, Lucie. Mobilní vzdělávací hry pro děti. *Učilek* [online]. 2024 [cit. 2024-03-31]. Dostupné z: https://ucilek.cz/android-hry-pro-deti/?fbclid=IwAR3Ppw7nidmDGfQrvMidWTWKxzzNHadWdXwSeuz3FZD25wRfcW11r4_fgrQ_aem_AVd0g1LFnziS4kEEwaqA1v_g_1--Dsmbpwb2fYEG86WHm4tI261weEuDuPVMfwnmskaUM7fvXhrhg8jEki01er0t
- (17) Matemág. In: *Mensa Česko: Pro nadané děti* [online]. 2024 [cit. 2024-03-31]. Dostupné z: <https://deti.mensa.cz/index.php?pg=tipy--hry--tablety-chytre-telefony&prid=234>
- (18) GAME REVIEW: LIGHTBOT. In: *Bold idea* [online]. [cit. 2024-03-31]. Dostupné z: <http://boldidea.org/blog/2016/7/22/game-review-lightbot>
- (19) PEŠEK, Ondřej. Co je to herní engine? *Zboží.cz* [online]. 2022 [cit. 2024-03-04]. Dostupné z: https://www.zbozi.cz/magazin/c/co-je-to-herni-engine/?_zn=aWQIM0QxMDgzNjMwNTYyNDM3NTgwMDI3JTdDdCUzRDE3MDY4ODgyNzQuMTczJTdDdGUIM0QxNzA2ODg4Mjc0LjE3MyU3Q2MIM0QwOEVCN0ZFRThFRDNGMzY1NjI0MjI1BRjc1OENENjA5Nw%3D%3D
- (21) BUTTFIELD-ADDISON, Paris. *Mobile Game Development with Unity*. Spojené státy americké: O'Reilly Media, 2017. ISBN 9781491944745.
- (22) Top 24 Mobile Game Engines and Development Platforms. *Capermint* [online]. 2014 [cit. 2024-03-04]. Dostupné z: <https://www.capermint.com/blog/top-20-mobile-game-engines/>
- (23) KRIEBERNEGG, Thomas. The best 22 mobile game engines and development platforms. *Appradar* [online]. [cit. 2024-03-04]. Dostupné z: <https://appradar.com/blog/mobile-game-engines-development-platforms>
- (24) VARAKSINA, Svitlana a Ivan DYSHUK. Mobile Game Development Process: Step-by-step Guide. *Mindstudios games* [online]. 2013 [cit. 2024-03-04]. Dostupné z: <https://themindstudios.com/post/mobile-game-development-process-step-by-step-guide/>
- (25) ALESSI, Patrick. *Vývoj her pro iPhone a iPad : programování pro iOS*. Brno: Zoner Press, 2012. ISBN 978-80-7413-199-8.
- (26) Question: Does the Choice of a Game Engine Matter in Game Development? In: *Dragonfly* [online]. [cit. 2024-03-06]. Dostupné z: <https://www.dragonflydb.io/faq/does-game-engine-matter>
- (27) Difference between mobile and pc game development. *Juegostudios* [online]. 2024 [cit. 2024-03-26]. Dostupné z: <https://www.juegostudio.com/blog/difference-between-mobile-and-pc-game-development>
- (28) Jak vytvořit Game design dokument. *Gamesdev* [online]. 2018 [cit. 2024-03-04]. Dostupné z: <https://gamesdev.cz/jak-vytvorit-game-design-dokument/>
- (29) UML Use Case Diagram Tutorial. In: *Lucidchart* [online]. [cit. 2024-03-26]. Dostupné z: <https://www.lucidchart.com/pages/uml-use-case-diagram>

- (30) Disciplína sběr a analýza požadavků. *Lucia Žoltá* [online]. [cit. 2024-03-26]. Dostupné z: <http://lucie.zolta.cz/index.php/softwareve-inzenyrstvi/150-disciplina-sber-a-analyza-pozadavku>
- (31) RYDVAL, Slávek. Můj život s UML. In: *Můj život s UML* [online]. [cit. 2024-03-26]. Dostupné z: <http://blok.kurzy-uml.cz/primarni-a-sekundarni-pomocne-pripady-uziti/>
- (32) Vývojový diagram - algoritmus. LASÁK, Pavel. *Jak na Microsoft Office* [online]. 2004 [cit. 2024-03-26]. Dostupné z: <https://office.lasakovi.com/excel/vba/vyvojovy-diagram-algoritmus/>
- (33) CHYTIL, Jiří a Zdeněk LEHOCKÝ. Vývojové diagramy - 1. díl. *Programujte.com* [online]. 2003 [cit. 2024-03-30]. Dostupné z: <http://programujte.com/clanek/2005080105-vyvojove-diagramy-1-dil/>
- (34) HARTINGER, David. Lekce 8 - Vývojové diagramy. *Itnetwork.cz* [online]. 2024 [cit. 2024-03-30]. Dostupné z: <https://www.itnetwork.cz/navrh/uml/vyvojove-diagramy>
- (35) DORAN, John P. *Unity 2022 Mobile Game Development*. 3. vydání. Velká Británie: Packt Publishing, 2023. ISBN 180461372X.
- (36) Kolik programovacích jazyků znáš, tolikrát jsi vývojářem. *Algotech* [online]. 2024 [cit. 2024-03-30]. Dostupné z: https://www.algotech.cz/novinky/kolik-programovacich-jazyku-znas-tolokrat-jsi-vyvojarem?fbclid=IwAR1ZVumN7PiA7Vuu18-7Nj5ZyEeyEUxjDogtxnntR1q2_ER_jLT6A3YAPXk_aem_ATa4xIN-Hh3b_xrpEuqgu0dWoPgastREIbDbIOzkbTGec-623wemOrWjg5So2DLMf--p5QvltSQSjdN3f_EuGs59
- (37) HURTA, Marián. Programuju, 2. díl: Jaký programovací jazyk si vybrat. *Engeto* [online]. 2016 [cit. 2024-03-30]. Dostupné z: https://engeto.cz/blog/programovani/programuju-2-jaky-programovaci-jazyk-si-vybrat/?fbclid=IwAR1QHZgtES8TJJxSBR87JohbW-nD0psbYzy_hSvjkb0mhE09T8Xn9Zx9y9A_aem_ATbi9urM9ubrhuKuqF7NRId0x5cR74H0XSMFeus83gt0C-nQDwGwxkspsg8J77JruhBIwQKAAXgx5CXbBzosPMY9v
- (38) ŠLAT, Martin. C# jako programovací jazyk her. *Robot_dreams* [online]. 2020 [cit. 2024-03-30]. Dostupné z: https://robotdreams.cz/blog/246-c-jako-programovaci-jazyk-her?fbclid=IwAR0GL2FJqycIIizhnYjJ83JwsGBairrcApkAeXqYifAIDt087SkCrX_GK2g_aem_ATa8IOxJW6h-cM1f8tDss3okWreCMTkcTCmpuVqqsv-6BYmlRHGL9cy6AoaqT9uu0uB2E3CpsqF9M0dc-DGVgHuZL

8 Přílohy

8.1 Ikonka hry



Příloha 1: Ikonka

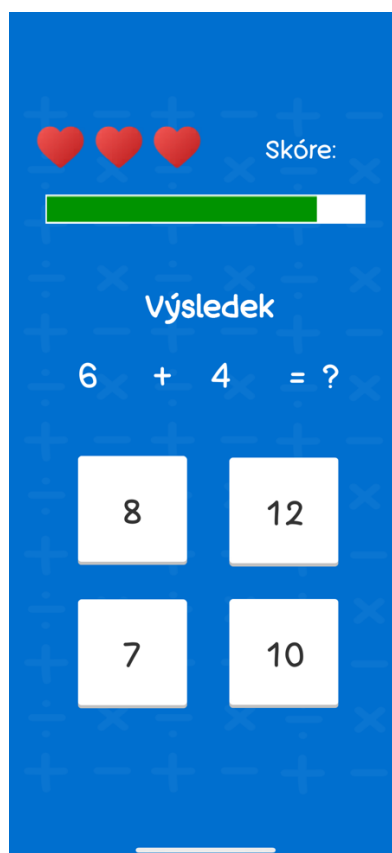
8.2 Scény hry



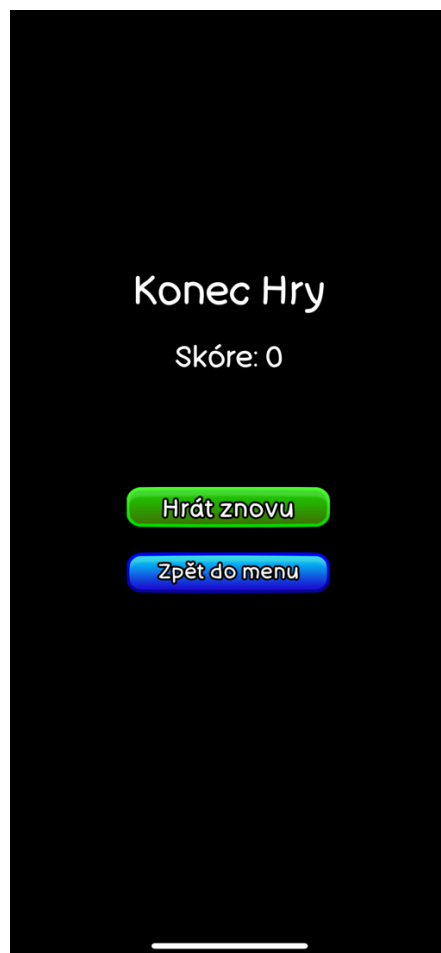
Příloha 2: Úvodní menu



Příloha 3: Nastavení



Příloha 4: Počítání příkladů



Příloha 5: Konec hry