



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

NEUMANN JÁNOS
INFORMATIKAI KAR



SZAKDOLGOZAT

OE-NIK	Hallgató neve:	Lakatos Márk
2024	Belső konzulens neve:	Dr. Tusor Balázs
	Hallgató törzskönyvi száma:	T009215/FI12904/N

SZAKDOLGOZAT FELADATLAP

Hallgató neve: **Lakatos Márk**
Törzskönyvi száma: T009215/FI12904/N
Neptun kódja: 

A dolgozat címe:

Mobil játék viselkedés modellezéssel
Mobile game with behavioral modeling

Intézményi konzulens: Dr. Tusor Balázs
Külső konzulens:

Beadási határidő: 2024. december 15.

A záróvizsga tárgyai: Számítógép architektúrák
Szoftvertervezés és -fejlesztés
Specializáció

A feladat

A hallgató feladata egy olyan telefonos játék/alkalmazás elkészítése, mely képes a játékos viselkedését modellezni, annak cselekedetei alapján. A játék inkrementális jellegű, követi annak a stílusát és játékmenetét, melyet a felállított felhasználói profil segítségével alakít annak érdekében, hogy a játékos preferenciáinak jobban megfelelő élményt nyújtson.

A dolgozatnak tartalmaznia kell:

- bevezetést,
- a megvalósításhoz felhasználható játékmotorok vizsgálatát,
- játékos modellezés szakirodalmát,
- a megvalósításhoz alkalmazott játékosmodell leírását,
- a megvalósított program leírását,
- a program működésének bemutatását,
- az eredmények ismertetését és értékelését.

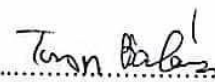



Dr. habil. Vámosy Zoltán
intézetigazgató

A szakdolgozat elévülésének határideje: **2026. december 15.**
(ÓE HKR 54.§ (10) bekezdés szerint)

A dolgozatot beadásra alkalmasnak tartom:

.....
külső konzulens


.....
intézményi konzulens

HALLGATÓI NYILATKOZAT

Alulírott hallgató kijelentem, hogy a szakdolgozat / diplomamunka saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült szakdolgozatomban / diplomamunkámban található eredményeket az egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja.

Budapest, 2024. 12. 11.



.....
hallgató aláírása



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Neumann János Informatikai Kar

KONZULTÁCIÓS NAPLÓ

Hallgató neve:

Neptun Kód:

Tagozat:

Lakatos Márk

Nappali

Telefon:

Levelezési cím (nl : lakcím):

Szakdolgozat / Diplomamunka¹ címe magyarul:

Mobil játék viselkedés modellezéssel

Szakdolgozat / Diplomamunka² címe angolul:

Mobile game with behavioral modelling

Intézményi konzulens:

Külső konzulens:

Dr. Tusor Balázs

Kérjük, hogy az adatokat nyomtatott nagy betűkkel írja!

Alk.	Dátum	Tartalom	Aláírás
1.	2024.03.07	Formai követelmények átbeszélése	Tusor Balázs
2.	2024.03.27	A játékos modellezés áttekintése	Tusor Balázs
3.	2024.04.19	Haladásról konzultáció	Tusor Balázs
4.	2024.05.09	Végző áttekintés	Tusor Balázs

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakdolgozat / Szakdolgozat I. / Szakdolgozat II. / Projektlabor 1 / Projektlabor 2 / Projektlabor 3 / Záródolgozati projekt / Diplomamunka I / Diplomamunka II / Diplomamunka III / Diplomamunka IV³ tantárgy követelményét teljesítette, beszámolóra / védésre⁴ bocsátható.

A konzulens által javasolt érdemjegy:5.....

Budapest, 2024.05.13.....

Tusor Balázs

Intézményi konzulens

¹ Megfelelő aláhúzendő

² Megfelelő aláhúzendő

³ Megfelelő aláhúzendő

⁴ Megfelelő aláhúzendő



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Neumann János Informatikai Kar

KONZULTÁCIÓS NAPLÓ

Hallgató neve:

Neptun Kód:

Tagozat:

Lakatos Márk.....

Nappali

Telefon:

Levelezési cím (pl.: lakcím):

Szakdolgozat / Diplomamunka¹ címe magyarul:

Mobil játék viselkedés modellezéssel

Szakdolgozat / Diplomamunka² címe angolul:

Mobile game with behavioral modelling

Intézményi konzulens:

Külső konzulens:

Dr. Tusor Balázs

.....

Kérjük, hogy az adatokat nyomtatott nagy betűkkel írja!

Alk.	Dátum	Tartalom	Aláírás
1.	2024.10.02	Szakdolgozat II átbeszélése	Tusor Balázs
2.	2024.10.16	Modellezés részleteinek átbeszélése	Tusor Balázs
3.	2024.11.11	Haladásról konzultáció	Tusor Balázs
4.	2024.11.18	Végső áttekintés	Tusor Balázs

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakdolgozat / Szakdolgozat I. / Szakdolgozat II. / Projektlabor 1 / Projektlabor 2 / Projektlabor 3 / Záródolgozati projekt / Diplomamunka I / Diplomamunka II / Diplomamunka III / Diplomamunka IV³ tantárgy követelményét teljesítette, beszámolóra / védésre⁴ bocsátható.

A konzulens által javasolt érdemjegy:⁵.....

Budapest, 2024.12.03.

Tusor Balázs

Intézményi konzulens

¹ Megfelelő aláhúzendő!

² Megfelelő aláhúzendő!

³ Megfelelő aláhúzendő!

⁴ Megfelelő aláhúzendő!

Kivonat

A felhasználó modellezés fontos szerepet foglal el az alkalmazások elkészítésében, viszont az elkészítésüknek a bonyolultsága és a köztudat hiánya miatt kevés cég alkalmazza ezt a stratégiát a legnagyobb cégeken kívül.

A szakdolgozat célja egy olyan program készítése, ami a játékos viselkedés modellezéssel dinamikusan változtatja egy játék menetét úgy, hogy a játékosnak minél élvezetesebb legyen. A cél, hogy a felállított profil használatával a játék dinamikusan változzon, a felhasználó által preferált cselekményeket helyezze előtérbe.

A program a Godot Engine nevű játékmotorban készül el, jelenetek és csomópontok összeillesztésével és a mögötte meghúzódó C# nyelven írt logika használatával.

Summary

User modeling plays an important role in the creation of applications, but due to the complexity of their creation and the lack of public awareness, few companies apply this strategy apart from the largest companies.

The aim of the thesis is to create a program that dynamically changes the course of a game by modeling the player's behavior so that it is as enjoyable as possible for the player. The goal is to make the game change dynamically by using the set profile, to prioritize actions preferred by the user.

The program is created in the game engine called Godot Engine, combining scenes and nodes and using the underlying logic written in C#.

Köszönetnyilvánítás

Ezúton szeretném kifejezni mély hálámat konzulensemnek, Dr. Tusor Balázsnak, aki a hasznos tanácsaival és szakértelmével hozzájárult a szakdolgozatom létrejöttéhez.

Továbbá szeretném megköszönni a szüleimnek és testvéremnek, hogy végig segítettek az évek folyamán és mindig megértéssel állnak mellettem.

Nem utolsó sorban, köszönetet mondok barátaimnak és kollégáimnak, akik bátorítással, jó tanácsokkal és segítséggel támogattak, miközben számos kihívás elé állítottak a dolgozat elkészítése során.

Mindezek az emberek nélkül nem sikerült volna ilyen színvonalas munkát végezni, és nekik köszönhetően tudtam teljesíteni a szakdolgozatom megírásával kapcsolatos feladataimat.

Tartalomjegyzék

Mobil játék viselkedés modellezéssel	2
1. Bevezetés.....	12
1.1 Kialakulás.....	13
1.2 Egy jó játék alapjai.....	13
1.3 Játékos modellezés.....	14
2 Irodalomkutatás és módszertanok feltárása	16
2.1 Alapfogalmak a videójátékiparban.....	16
2.2 Inkrementális játékok	17
2.3 Az inkrementális játék felépítése meghatározó jellemzői.....	18
2.4 Játékmotor választás	20
2.4.1 Lehetséges játékmotor opciók	20
2.4.2 Godot Engine	22
3 Követelmény specifikáció.....	24
3.1 A szoftver használata	25
4 Fejlesztői specifikáció.....	26
4.1 A nyersanyag tároló jelenet	27
4.2 Kreditek fejlesztése.....	27
4.3 A segítő.....	28
4.5 Valuta váltó	29
4.6 Játékos modellezés.....	29
5. A szoftver megvalósítása	31
5.1 A nyersanyag tároló megvalósítása.....	31
5.2 A kreditek fejlesztésének megoldása	33
5.3 A segítő.....	34
5.4 A játékos modellezés megvalósítása.....	35
5.5 Valuta váltó	35
6. Eredmények	37
6.1 Termék orientált játék eredménye.....	37
6.2 Segítő orientált játék eredménye.....	38

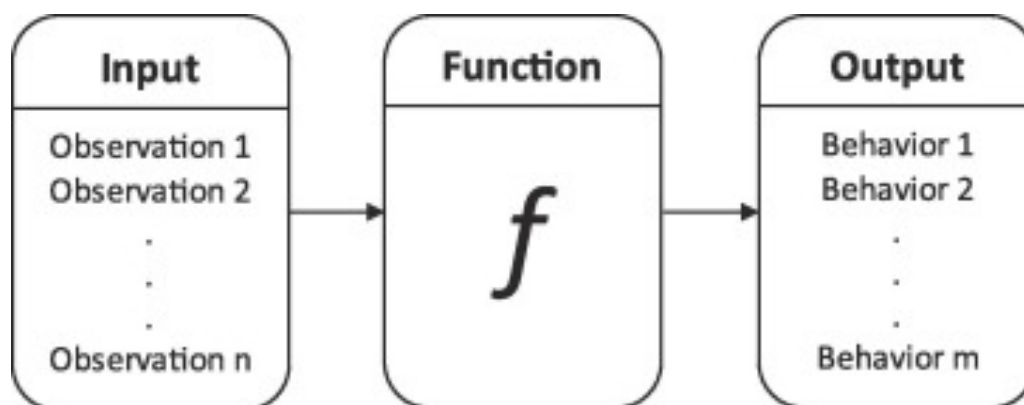
6.3 Kredit és valuta orientált játék eredménye	38
7. Összefoglalás	40
8. Summary	41
9. Irodalomjegyzék	42
10. Ábrajegyzék	43

1. Bevezetés

A mobiltelefon a modern kori életnek a természetes része. 2022-ben a 30 év alatti felnőttek közel 100%-a használt internetet és okostelefont. Az évek alatt újabb és újabb hasznos funkciókkal lettek felruházva a szimpla kommunikáción kívül, ezzel szinte minden ember találhat benne számára hasznos eszközöket. Ezek közül a legfontosabb az internet elérése, ami a 2000-es évek elején jelent meg először. Az internet elérésével nem csak egy végtelen tudásbázissá változtatták a telefont, hanem nagyban megkönnyítették az online vásárlást, szórakoztató anyagok elérését és a hívás nélküli kommunikációt.

Az okostelefon elterjedésével együtt jöttek rá a játékok is. Míg korábban a számítógépes és a konzolos videójátékok voltak a digitális játékok meghatározó formái, a mobiljátékok vitathatatlanul a kortárs, domináns jelenséggé váltak. Ez a fejlődés meglehetősen gyors volt - mindössze néhány évtized alatt zajlott le és átalakította mind a játékok tartalmát és gyakorlatát, mind pedig a játékvilágot. 2022-ben a videójáték ipar bevételének 53%-át a mobil játékok tették ki és csak Androidon közel fél millió játék közül választhat a játékos. A világ egyik leghíresebb játéka, a Candy Crush 2.7 milliárd letöltésen állt 2020-ban.

Ahogy a játékok egyre nagyobbak és komplikáltabbak lesznek, egyre nagyobb lesz a szükség arra, hogy a fejlesztők megtalálják és megértsék, hogy a játékosok mit élveznek a programjaikban. A játékos modellezés, konkrétan a játékos a környezetével és más játékosokkal való viselkedésének absztrakt leírása. Ezt az információt felhasználva, a program dinamikusan változtathatja az elemeit, hogy az jobban megfeleljen vagy nagyobb kihívás legyen a felhasználónak. A modellezés folyamata ábra mutatja, hogy egy program, hogy gyűjti ki a játékosok viselkedését és kategorizálja őket profilokba. Ennek a szakdolgozatnak a célja, egy olyan játékos modellező készítése, amely képes a játékos viselkedése alapján egy modellt készíteni róla majd, a modell alapján dinamikusan változtatni a játékmenetet.



1. ábra: A modellezés folyamata

1.1 Kialakulás

Az eddigi játékokhoz szükséges fix helyzetet leváltotta a kézben hordozás lehetősége, ezzel behozva a játékokat a mindennapokba. Természetesen az emberek előtt nem volt ismeretlen a magaddal hordozható játékok fogalma, viszont ezek nem digitálisak voltak. Az elektronika miniaturizálása gyorsan haladt, és már az 1970-es évek végén megjelent az egyszerű, kézi elektronikus játékok első generációja. Ilyen volt például a Mattel Auto Race (Mattel, 1976) és a Merlin (Parker Brothers, 1978). Valószínűleg a legnépszerűbb termékcsalád az egycélú, hordozható elektronikus játékeszközök evolúciójában a Game & Watch volt, ami eredetileg a Nintendo gyártott 1980 - 1991 között.

A mobilos játékok népszerűsítése a 90-es évek végén kezdődött a Snake, Nokia telefonokra való előre letöltésével. Ennek sikerességét követve, több cég is előre telepített játékokat helyezett az eszközeire. Az első igazán komoly készülék az iPhone volt, ami teljesen megváltoztatta a mobilos játékok irányát. A mobiljátékok tervezésének korai fázisában gyakran a meglévő videójátékok egyszerűsítésére volt a cél, főként a rendelkezésre álló számítási kapacitás, teljesítmény és a memória korlátjai miatt. A kutatás és a fejlesztés előrehaladtával egyre nyilvánvalóbbá vált, hogy a mobiljátékoknak is egyedi erősségei vannak, amelyeket más játékkplatformok nem tudtak biztosítani.

A legtöbbet használt platform vitathatatlanul a mobil. Azzal az egyedi címmel büszkélkedhet, hogy ez az egyetlen platform, amivel bármely generáció több mint 50%-a játszik. A Z generáció (1995-2009 között születettek) körében 81%-a vesz rész a mobil játékokban, míg a Baby Boomer generáció (1946-1964 között születettek) 70%-a.

1.2 Egy jó játék alapjai

Nehéz meghatározni mi tesz ki egy jó játékot. Nem egyszer megtörtént, hogy hasonló karakterisztikákkal rendelkező játékokból az egyik sikeres lett, míg a másik megbukott. Talán az egyik legfontosabb tulajdonsága egy játéknak, hogy a játékos mennyire tud benne elmerülni. Ha egy játékos el van merülve a játékban, elveszti az időérzését és abszolút az előtte lévő feladatra koncentrál. Ezt az állapotot előhozhatja egy jól kifizetődő feladat azonnali visszajelzésekkel és világos célokkal. Emellett sokat segít, ha a játékos úgy érzi nála a kontroll, ő uralja azt, hogy mi történik a karakterével. Az 1980-as évek elején Malone [1] meghatározott három kritériumot egy élvezetes játék létrehozásához. Ez a három kritérium a kihívás, fantázia és a kíváncsiság. Azóta sok minden adódott ahhoz, hogy milyen is egy jó játék, de ezek a kritériumok továbbra is részét képezik annak.

A mai játékok sokkal összetettebbek, mint a 80-as években, ebből adódóan több kritériumnak is kell megfelelniük, hogy élvezetesek legyenek. A jó történet mesélésnek hasonló feltételei vannak, mint a filmeknek. Eléggé érettnek kell lennie, hogy lefoglalja az idősebb játékosokat is. Fent kell tartania a feszültséget, hogy el lehessen merülni benne és megfelelően kell, hogy illeszkedjen a játékmenetbe. A játékmenetnek kiegyensúlyozottnak, igazságosnak és innovatívnak kell lennie. Fontos, hogy a tanulási görbe ne legyen túl meredek, könnyen el lehessen kezdeni, de nehéz legyen profi módon elsajátítani. Egy problémára jó, ha többféle lehetséges megoldás van biztosítva, a játékos ezzel úgy érzi ő irányítja a történéseket. Ezen felül nem szükséges minden részt a játékban kötelezővé tenni, amennyiben a játékos szeretne a fő történet szálán gyorsan végig haladni legyen meg rá a lehetősége. A legújabb játékok nagyon fejlett játékmotorokban készülnek. Ezekbe bele van építve rengeteg funkció a fizikától kezdve a fény visszatükröződésig, ami abban segít, hogy minél realisztikusabban nézzen ki a játék. Ezek a motorok nem csak a kinézetben segítenek, hanem a hangeffekteknél is. Az, hogy más felületeken máshogy hangzik a karakter lépte, egy barlangban visszhangoznak a leeső kövek mind apró részletek, de az egész világot természetessé teszik, nem törik meg a játékos beleélését.

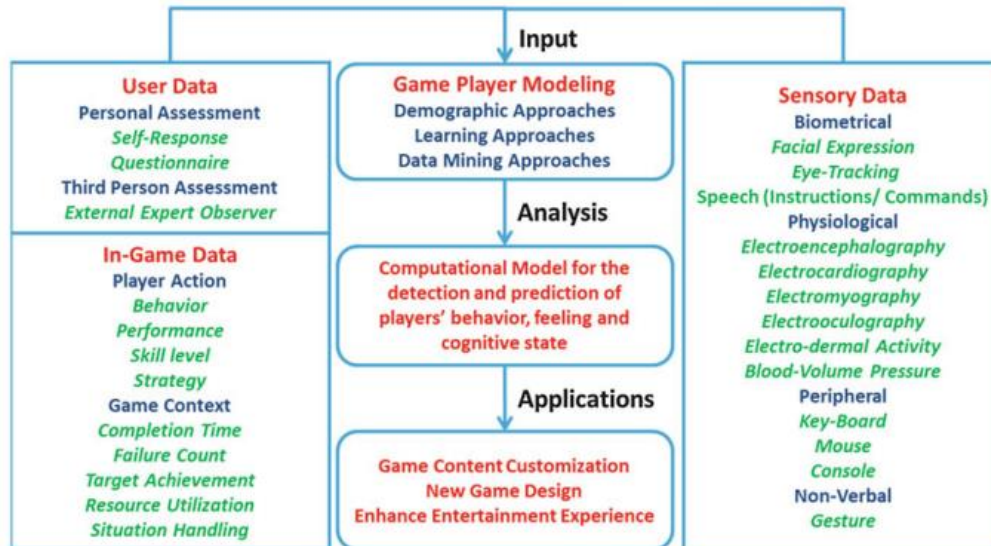
1.3 Játékos modellezés

A játékos modellezés egy absztrakt leírást képez a játékosokról, számítási modelleken keresztül. A modell segít felismerni, kifejezni és megjósolni az érzéseit és viselkedését a játékosoknak ezzel lehetőséget adva személyre szabni a játék élményt, vagy felfedezni olyan részeit a játéknak, amit a játékosok nagy része frusztrálónak vagy egyszerűen rossznak talál. Ezen kívül a fejlesztők tudnak információt gyűjteni arról, hogy milyen mechanikákat, feladatokat vagy történet elemeket találnak a játékosok élvezetesnek így tudják a jelenlegi és következő játékok fejlesztésének irányát meghatározni.

A játékos modellezés első példája az 1970-es években történt meg, ahol Slagle és Dixon megpróbálta modellezni az ellenfelek viselkedését klasszikus játékokban a hibáik alapján [2]. Manapság a játékos modellezés az ellenfél viselkedésének elemzéséről inkább áttért más témák kutatására, mint például a játékos elégedettsége [3]. Továbbá lehetséges a futásidő alatti nehézség változtatása, különböző stratégiák és játék béli stílus elemzése.

Egy modellnek három különböző inputja lehet. Felhasználói adatok, érzéki adatok és játékon belüli adatok. A felhasználói adatok magukba foglalják a személyes megítélést és külső megfigyelést. Néhány elhanyagolható korláttal rendelkezik ez az adatfajta, mint nem releváns adat értékelések és rövid távú memória. Az érzéki adatok a játékos testére felszerelt szenzorokból nyerhető ki. A leggyakoribb szenzor adatok közé tartozik a biometrikus, fiziológiai, és perifériás. Ezek a szenzorok pontosság és teljesítmény terén még több kihívásba ütköznek. A játékon belüli adatok a játékon belül végrehajtott tevékenységek, készségek, teljesítmény és viselkedés. Ezen kívül lehet mérni a szint teljesítésének idejét, az erőforrások

kihasználását, az elkövetett hibák számát és a helyzetkezelést. A kihívás ezzel, hogy az adat megfelelően legyen értelmezve magas szintű modellezésre, limitált mennyiségű adattal. A játékos modellezés különböző bemenetei ábra levezeti, hogy a három inputot milyen módon lehet begyűjteni, előnyeit és hátrányait.



2. ábra: A játékos modellezés különböző bemenetei

Az elmúlt években nagy számú sikeres, ingyenesen játszható játékok jelentek meg, különböző platformokon. Mivel maga a játék ingyenes, a bevétel a játékon belüli mikro tranzakciókból jön, amivel extra tartalmakat tudnak a játékosok feloldani. Azért, hogy ezek a játékok jövedelmezőek legyenek a fejlesztőknek folyamatosan meg kell figyelniük a játékosok vásárlási szokásait. A játékos modellezése javíthatja a játékosok cselekedeteinek megértését és megkönnyítheti azoknak a játékosoknak az azonosítását, akik készek fizetni. Ezenkívül a játékosok modellezési adatai felhasználhatók a piaci eljárások és tartalom testre szabására, ami végül magasabb jövedelmet eredményezhet.

2 Irodalomkutatás és módszertanok feltárása

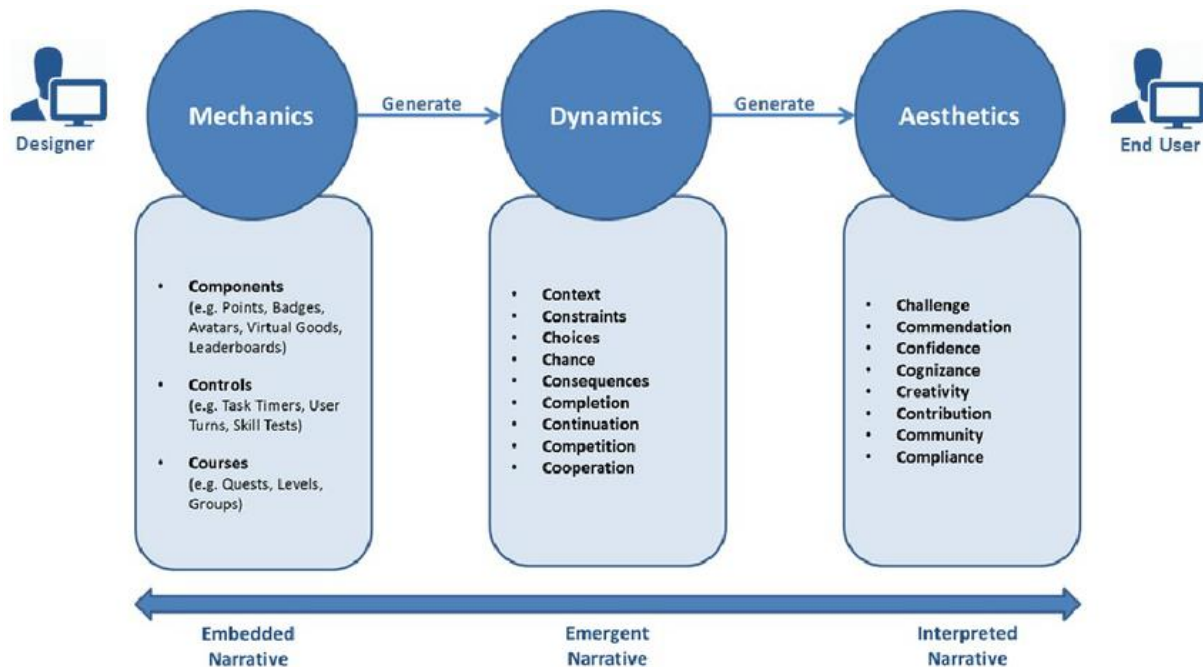
Az irodalomkutatás bármilyen szoftver elkészítése előtt egy fontos lépés. A hasonló módszertanok és rendszerek ismeretével jól informált döntéseket lehet hozni, ihletet meríteni és hatékonyabbá tenni a programot a fejlesztés során.

2.1 Alapfogalmak a videójátékiparban

- **Játékkonceptió:** A játék létrehozásához először egy érdekes és eredeti koncepcióra vagy ötletre kell gondolni. Ezenkívül ki kell választania, hogy milyen játékot szeretne készíteni. Ezenkívül lehetnek tervei a cselekményre, egyedi elemekre és esztétikára vonatkozóan. A játékterv dokumentuma az a terület, ahol ezeket a koncepciókat dokumentálni kell.
- **Játékmechanika:** A játék ezen struktúrák és szabályok miatt működik. Meghatározzák, hogy mit lehet csinálni a játékban. Ezen kívül a játék akadályait, eredményeit, azt, hogy lehet a karaktert irányítani. (Harcrendszer/vezetés)
- **Játéktervezési elvek:** Egy játék elkészítéséhez lehetséges néhány "szabályt" követni, amik segítik a fejlesztőt nem elveszni a lehetőségek és részletek halmazában. Például fontos, hogy egy fő játékmechanika tökéletesítése legyen a cél, mivel ezek nagyon bonyolultak, több potenciálisan rontja a minőségüket. Játéktervezési elvek ábrán ezek az elvek és a megfelelő implementálási sorrendjük látható.
- **Játék motorok:** A játékmotor egy szoftver keretrendszer, amelyet elsősorban videójátékok létrehozására használnak. Általában olyan támogató alkalmazásokkal érkezik, mint a szintszerkesztő és más releváns könyvtárak.[4] A játékmotorok alapvető jellemzői általában a 2D vagy 3D grafikákhoz használható renderelő motorból (más néven "renderelőből"), fizikai motorból vagy ütközésészlelésből és -reakcióból, hangból, szkriptelésből, animációból, mesterséges intelligenciából, hálózatiépítésből, streamelésből továbbá a memóriakezelésből állnak.
- **Célok:** Ahhoz, hogy a játékosoknak legyen irányuk és céljuk a játékban, a játékoknak jól meghatározott célokkal kell rendelkezniük.
- **Egyensúly és nehézség:** A játékoknak megfelelően nehéznek kell lenniük, hogy kihívás is legyen, de nem is lehetetlen. A játékosok motiváltságának megőrzéséhez meg kell találni az ideális egyensúlyt. Azt, hogy mindenki megtalálja a neki megfelelő nehézséget általában különböző nehézségi szintek bevezetésével segítik.
- **Haladás és jutalmak:** A játékokban gyakoriak az olyan rendszerek, amelyek lehetővé teszik a játékosok előre jutását és nyereményeket játék közben. Ezek közé tartozik az erő megszerzése, az új szintek elérése és az exkluzív tárgyak beszerzése a célok

teljesítéséhez.

- **Visszacsatolás:** A játékosok visszajelzést kapnak a játékoktól a vezérlés, hangok és grafikák révén. Ezenkívül a játékkervezők odafigyelnek a játékosok megjegyzéseire, és javaslataikat beépítik a későbbi verziókba.
- **Narratíva:** Történetek elmesélésével, lenyűgöző karakterek fejlesztésével és gyönyörű, élethű környezet kialakításával a videójátékok magával ragadó élményeket teremtenek. Ez az az érzés, hogy a játékok világának része.



3. ábra: Játékkervezési elvek

2.2 Inkrementális játékok

Az inkrementális játékok, más néven „ambient, idle, clicker” nagy népszerűséget nyert az elmúlt években. [5] Manapság általában mobilon játsszák, de továbbra is el lehet érni őket PC-n és az interneten. A játékban az előrehaladást többnyire minimális vagy semmilyen interakcióval lehet elérni.

Az olyan mechanikák, amik többször is megjelennek és hatással vannak az összes többi alrendszerre központi mechanikának nevezzük. A központi mechanika, vagyis a játék leggyakrabban visszatérő cselekvései a nyersanyagok gyűjtésére alapul, amit egy egyszerű, ismétlődő lépésekkel lehet elérni. [6] A legtöbb játék (ebben a stílusban) tartalmaz olyan mechanikákat, amelyek automatizálják a játékmenetet, így a játék magától halad.[7]

Az erőforrás felhalmozás és a kiadások kezelése kritikusan fontos része a

játkegyensúlynak. Ha gyorsabban halmozódnak fel az erőforrások, mint amennyit el lehet költeni, a játékos soha nem fog hiányt tapasztalni így, nem kell nehéz döntéseket meghoznia. Ha a játékos soha nem tud elég erőforrást keresni ahhoz befolyásolja a játékot, nem tud rendesen haladni vele, így hamar megunja. Az erőforrás-termelési ráták és a költségek egyensúlya, az inkrementális játékok kritikus eleme. [8] Bármennyire is egyszerűek ezek az interakciók, a játékosok továbbá is pozitívan reagálnak ezekre.

A legtöbb digitális játékkal ellentétben az inkrementális játékoknak egy alap játékeleme az, hogy nem játsszák, mivel amíg a játékos távol van a játék tovább játszódik, gyűjti a nyersanyagokat. Ez a stílus elsősorban egy paródiaként kezdődött, de 2013-ban egy "Cookie Clicker" nevű játék behelyezte a köztudatba. Ez a játék egy letisztult animált, rendkívül gazdag grafikus felületet kínál: a játékos elkezd egy süti képére kattintani, ezzel gyűjti a nyersanyagot. Kattintásonként egy sütit szerez meg, amelyeket karakterek és épületek vásárlására költhet el ezzel növelve a nyersanyag automatikus gyártási sebességét.

Az inkrementális játékoknak több alfaja is van:

- A nulla-játékos játékokat formálisan olyan játékokként határozzák meg, „amelyekre nincs szükség emberi közreműködésre”. [9] Ezeknek a játékoknak négy altípusa van, beleértve a játékokat, ahol a játékos csak a beállítás során avatkozik be és olyan játékok, amelyek teljesen önmagukat játszanak.
- Az ambiens játékok, amelyek elnevezése az így elnevezett zenei stílusból származik, csökkenti a játékos interakcióját azáltal, hogy a háttérben marad, miközben a játékos más dolgokat csinál. [10] Ezekben a játékokban a játékos könnyedén válthat a játék és egy feladat között mivel a játék csak perifériás figyelmet igényel.
- Végül a háttérjátékokat olyan játékokként határozzák meg, „ahol a munka nagy része akkor történik, amikor a játékos nem játszik”.

2.3 Az inkrementális játék felépítése meghatározó jellemzői

Ahhoz, hogy egy játék inkrementálisnak legyen minősíthető, több kritériumnak is meg kell felelnie.

- a játék nagy része a háttérben játszódik
- ezekben a játékokban várakozás egyenlő a játékkal, a játék előre haladhat miközben a játékos nincs jelen, gyakran a mechanikán keresztül
- amelyek automatizálják a játékmenetet
- időbeli rugalmasságot biztosít – a játékosok nyugodtan félretehetik a játékot, és bármikor visszatérhetnek hozzá mindenféle hátrány vagy büntetés nélkül
- gyakran elkezd a játékos egy játékot és azt játssza éveken keresztül, azaz nem csinál új karaktert, fiókot
- nincs a játéknak vége, végtelenségig játszhatja a játékos, ha szeretné

- van a játék újrakezdésre lehetőség, újult különleges képességekkel

Azt is figyelembe kell venni, hogy milyen mennyiségű interakciót tartalmaznak ezek a játékok. Az interaktivitás ebben a stílusban nem kizárólagos kategóriák szerint értelmezhető, hanem az interaktivitás fokozataiként egy spektrum mentén, azoktól a játékoktól, amelyek nem igényelnek más interakciót, mint a játék elindítását addig amelyek rendszeres játékos részvételt igényelnek. Míg egyesek játékok egy helyen maradnak a spektrumon, jellemzőbb, hogy a játék előrehaladtával mozog a spektrumon.

Három gyakori minta található meg:

- **Kattintó:** A spektrum magasabb interakciós végén a klikkeres játékok állnak. Ezek a játékok kattintással, dörzsöléssel vagy koppintással járnak, ezzel károkat okoznak és/vagy erőforrásokat generálnak egy objektumon végzett többszöri kattintással, amelyeket várakozási idők választanak el egymástól. Néhány ilyen játék ezzel kezdődik, ameddig fel nem tud a játékos különböző felszereléseket, tárgyakat feloldani. Ezzel általában elhalványulnak ezek a mechanikák, miután tovább haladnak a játékban. Más játékokban a játékosnak továbbra is kell kattintania, hogy rendszeresen haladhasson, viszont a kattintási gyakoriságot széthúzzák hosszabb várakozási időre. Az "Egy inkrementális játék kinézete" ábrán a leghíresebb kattintó játék látható, a Cookie Clicker.
- **Minimalista:** A minimalista játékok a rendelkezésre álló interakciók számát erősen lecsökkentik akár a játékmechanikán keresztül, amelyek automatizálják a játékmenetet, vagy a játékmenet fázisait, amelyek csökkentik a játékos interakciójának lehetőségét.[11] Ezek a játékok tartalmazhat játékos interakciót nem igénylő részeket.
- **Zéró-Játékos:** A spektrum másik oldalán a zéró-játékos játékok állnak. Ezek játékok nem igényelnek semmilyen játékos közreműködést a kezdés után, vagy lehetővé teszik korlátozott beavatkozást a beállítások során, de nem befolyásolhatják a játékmenetet.[12] Ezek a játékok másodlagos elfoglaltság jellegűek, a korlátozott interaktivitás és a háttérben nyitva hagyás lehetősége miatt, készen arra, hogy a játékos bármikor foglalkozhasson velük.



4. ábra: Egy inkrementális játék kinézete

2.4 Játékmotor választás

A játékmotor egy olyan szoftver, amely az összes szükséges összetevőt egyetlen fejlesztői környezetbe egyesíti a játékok létrehozásához. A professzionális játékmotornak elegendő funkcionalitást kell tartalmaznia ahhoz, hogy a játék gyártási folyamatának minden szakaszában megfeleljen, a kezdeti prototípusoktól a végső polírozásig. Figyelembe véve mennyi lehetőség van fontos, hogy olyat válasszunk, ami a fejlesztői igényeinknek legjobban megfelel.

- A játék típusa: Olyan tényezőket kell itt figyelembe venni, mint a választott vizuális stílus, a játék műfaja, a játék mechanikájának összetettsége, 2D vagy 3D grafika és még sok más.
- Cél platform: Egyes játékmotorokat úgy tervezték, hogy megkönnyítsék a mobil eszközökre való fejlesztést, míg másokkal több platformra lehet fejleszteni, és van olyan is, ami kiváló abban, hogy egyetlen platformra koncentráljon.
- A fejlesztői csapat tudása, tapasztalata: A játékmotor kiválasztása megköveteli a fejlesztőcsapat tapasztalatának alapos mérlegelését, mivel egy új motor elsajátítása nehéz lehet mind az egyének, mind a fejlesztői csoportok számára.
- Költségvetés: A játékmotor-licenck ár változó, ezért figyelembe kell venni gondolni mekkora a fejlesztői csapat költségvetése.

2.4.1 Lehetséges játékmotor opciók

Unreal Engine: Az egyik leghíresebb és legtöbbet használt motor. Lényegében ez egy többplatformos, játékfejlesztő motor, bármekkora méretű vállalkozás számára. Az eredeti verzió 1998-ban jelent meg és azóta is a leghíresebb játékok közül nem egy ezt használta. Az

erőssége abban rejlik, hogy eléggé lehet módosítani ahhoz, hogy a játékból egészen egyedi élmények legyenek. A hátránya a motornak, hogy a piacon a nagy nevek közül ezt a legnehezebb megtanulni.

Unity: Egy többplatformos motor, amivel könnyedén lehet létrehozni interaktív 3D tartalmakat. A minden az egyben szerkesztőjével az összes elképzelhető platformmal kompatibilis. Felhasználóbarát felülete van, ami nagyban megkönnyíti a képzést. Hátránya, hogy viszonylag könnyű egy kódot futtatni még ha a kód nem is túl jó ezzel sok optimalizálási problémát okozva. Emellett nem túl rugalmas ezért a nagyobb méretű játékokhoz nem a legjobb opció.

Godot: A Godot ingyenesen használható az MIT licenccel keresztül nyílt forráskódú. Nincs jogdíj, nincs előfizetési díj. A motor kiváló 2D és 3D fejlesztésre is. Nagy általános eszközkészletet kínál, így a legtöbb problémára már van kész megoldás. Mivel nyílt forráskódú, a közösség folyamatosan javítja, fejleszti. Előnye, hogy alacsony a gépigenye, egyszerűbb eszközökön is lehetséges használni. Kevés hátránya között van, hogy konzolokra nem lehet a motoron fejleszteni és hiába van nagy eszközkészlete a komplikált részekben nem ragyog.

GameMaker: Studio: A GameMaker széles körben elterjedté vált, mert nem igényel programozási ismeretet. A fejlesztő előre elkészített kódblokkokat köt össze, ezzel fűzve össze a programot. Ezzel sokkal gyorsabban lehet programot létrehozni, mint az anyanyelvi kódolással. Ezekkel a típusú motorokkal az a probléma, hogy a fejlesztők sokkal korlátozottabbak.

Az különböző játék motorok előnyei és hátrányai ábrán számokkal is ki van fejezve, hogy melyik motornak melyek az erősségei és gyengeségei.

	Installation & Ownership	2D/3D	Ease of Use	Integration & Compatibility	VR Support	Customer Support
Unreal Engine	***	Both	***	*****	Yes	***
Amazon Lumberyard	***	3D Only	*****	***	Yes	***
CryENGINE	***	Both	***	*****	Yes	***
Unity	***	Both	***	*****	Yes	***
GameMaker: Studio	*****	2D Only	*****	***	No	***
Godot	*****	Both	*****	***	No	*****
Cocos2d	*****	2D Only	*****	***	No	***

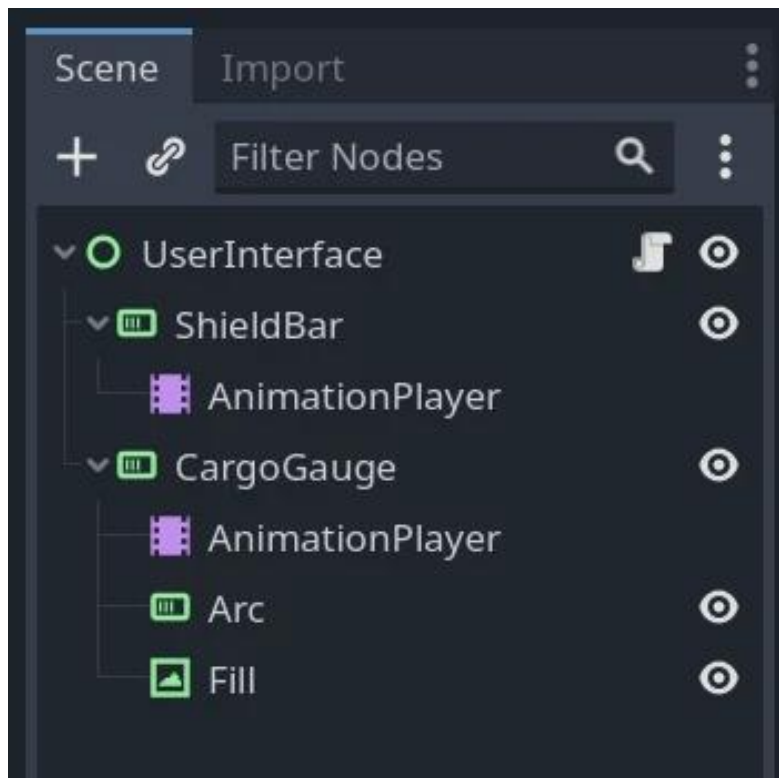
5. ábra: Különböző játék motorok előnyei és hátrányai

2.4.2 Godot Engine

2014-ben Juan Linietsky és Ariel Manzur által megjelent a Godot. A Godot egy platformfüggetlen játékmotor, ami 2D és 3D játékok fejlesztésére is használható. A játékmotor a fejlesztést erősen elősegíti eszközök széles skálájával. Tartalmaz grafikus renderelő motort, hanglejátszó eszközöket, animációs eszközöket és még sok mást. 2014 óta a motor rengeteget lett fejlesztve, a közösség által elősegítve. A motor számos olyan funkcióval rendelkezik, ami a 3D grafikához szükségesek.

Ezek között van a világítási rendszerek például fénytörés és árnyékolás, fizika, mint gravitáció és objektumok ütközése. Más motorokkal ellentétben a Godot valódi, dedikált 2D motorral rendelkezik. Más motorok ezt a Z tengely ellapításával érik el, a 2D illúzióját keltve. Ennek a következménye, hogy a motor hatékonyan tudja kezelni a 2D háttérszámításokat, pixel alapú egységeket. Ezen kívül rendelkezik dedikált rendszerekkel, mint a “csempetérkép” szerkesztők, 2D fizika, világítási rendszerek.

A rendszer erősen objektum-orientált tervezés elvére épül, ennek elsődleges módja a csomópont és jelenet architektúra. A Godot Engine jeleneteinek felépítése ábrán ennek a felépítése látszik. Minden objektum jelenetként van meghatározva és minden jelenet több csomópontot tartalmazhat. Több száz különböző csomópont van legyen szó tárolókról, felugró ablakokról, fizikai elemekről, animációkról, hangokról, a felhasználó számára készített elemekről, mint jelölőnégyzetek és gombok. Minden egység amire szükség lehet egy bonyolultabb mechanika kialakításához megtalálható csomópontként. A csomópontokat rengeteg módon lehet igazítani a fejlesztő igényeihez, az egyik legfontosabb ilyen elem lehetővé teszi, hogy minden csomópontnak saját textúrát adjunk és ezt futásidőben változtathassuk is. Egy jelenet egy komplexebb részét jelenti a programnak, amit a fő megtekintésben lehet integrálni. Egy jelenetet többször is újra lehet használni egy szülő jelenetnek a kódjában példányosítva. A jelenet változóinak értékeit adva más-más viselkedésre lehet bírni őket, például a jeleneten belül van egy logikai változó, ami meghatározza, hogy egy másik jelenetből érkező kérés válthat e rá ki kölcsönhatást. Ennek a funkciónak az az előnye, hogy jól lehet szabályozni a jelenetek közötti öröklődést és más motorokhoz képest sokkal könnyebben kezelhetővé a példányokat. A Godot rengeteg platformra képest importálni, mint Windows, macOS, Linux, iOS, Android. Ezen kívül lehetséges a HTML5 exportálás arra az esetre, ha a fejlesztő az internetre szeretné kirakni a játékát. Egyetlen hiánya, hogy konzolokra jelenleg nem kínál támogatást.



6. ábra: Godot Engine jeleneteinek felépítése

A motor lehetővé teszi a fejlesztők számára, hogy különböző nyelven kódoljanak. Van egy saját nyelve, a GDScript ami kifejezetten a motorral való együttműködésre van kifejlesztve. Szorosan integrálódik a rendszerrel magával ezért ez az egyik legoptimálisabb lehetőség. A GDScript sok tekintetben hasonlít a Pythonhoz, tiszta és ember által olvasható. A kezdők számára létezik egy kódblokk összekötő, vizuális kód készítő, amivel egyszerű parancsokat tudnak összefűzni. Továbbá támogatva van a C++ és C# is. A Godot GDNative API-ja támogatja a korlátozott nyelvi kötést így lehetséges olyan nyelveket is használni, mint a Ruby vagy R. A több nyelven kívül meg lehet más alkalmazásokban is nyitni a kódot, mint például a Visual Studio, ahol több segítő rendszer van a gyors kódoláshoz.

A Godot egy nyílt forráskódú motor, azaz a forráskódja nyilvánosan elérhető. Ez azt jelenti, hogy a fejlesztők nem csak terjeszthetik, hanem saját belátásuk és képességük alapján, megkötések nélkül módosíthatják azt. Ez azt is jelenti, hogy közösség vezérelt, szabadon hozzá lehet járulni a motorkód fejlesztéséhez. Továbbá teljesen ingyenes nem úgy, mint más hasonló motorok.

3 Követelmény specifikáció

Egy inkrementális mobiljátékot tervezek iOS és Android készülékekre, „The Bar” néven. Egy olyan játékot tervezek, amellyel az emberek több hónapon keresztül tudnak játszani. A játék célja, hogy a játékos elmerüljön az inkrementális játékok varázsában és ha egy kis szünetet szeretne tartani munka vagy tanulás közben, ez a játék kösse le. A program egy egyedi történetben elhelyezett üzletvezetőről szól, ahol a játékos nyersanyagokat gyűjt, hogy tovább tudja fejleszteni az üzlet helységét. Az egyediségét az adja, hogy ilyen történettel, még nem rendelkezik inkrementális játék. A stílus alapvető elemei megtalálhatóak a játékban, mint a nyersanyag gyűjtés, az exponenciálisan növekvő árak és az előrehaladás újrakezdése, jobb eszközök érdekében. Ezen kívül több tartalom is el lesz helyezve, hogy megtörje a monotonitást.



7. ábra: Exponenciálisan növekvő termék ár

A játék a Godot Engine-ben fog elkészülni. Több indokból is esett erre a választásom, dedikált 2D motorja van, ami tökéletesen illik ehhez a stílushoz, használni lehet benne a C# programozói nyelvet, ami az elsőrendű nyelvem. Felhasználó barátságos egy kezdőnek könnyebb elkezdni benne dolgozni.

Végül lesz a programban egy játékos modellező is elhelyezve, ami viselkedés alapján különböző kategóriákba helyezi el a játékost. Lesz 3-4 más-más pontrendszer, amihez a játékos cselekménye alapján hozzáadunk vagy kivonunk. Ezzel meg lehet tekinteni a játékos milyen arányban húzódik egy fajta cselekvési forma irányába.

3.1 A szoftver használata

A játékos először egy nyersanyaggal kezd és ha eleget gyűjt össze belőle, egy újabbat is feloldhat. A nyersanyag megszerzése lehet manuális, vagyis a játékos saját akciója alapján készül el vagy lehet automatizált, ami megszerezhető segítők gyártják előre megadott időközönként.

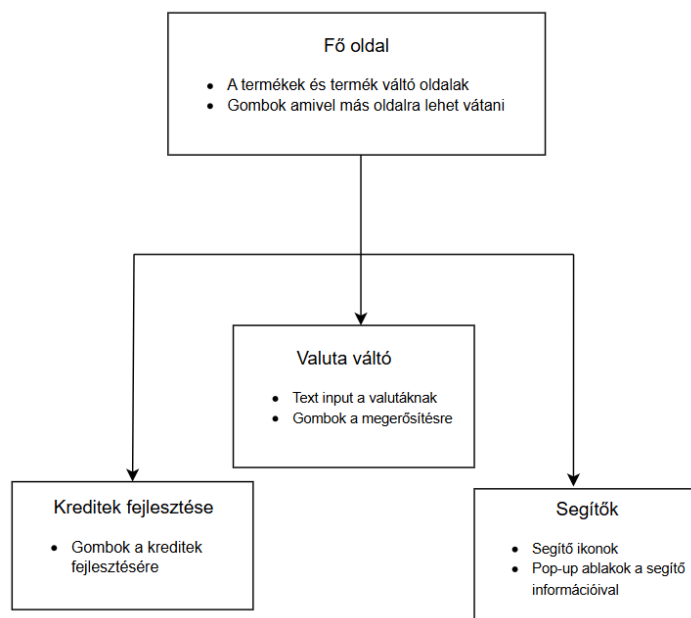
Három különböző fizetőeszköz lesz a játékban: kredit, segítőpontok és arany. A kredit az alapvető pénz a játékban. Nyersanyaggal együtt ezzel lehet új dolgokat vásárolni. Ha elér az üzlet helység bizonyos termék mennyiséget, segítőpontokat kap a játékos. A segítők egy nyersanyagot fognak termelni vagy krediteket, a segítőket a segítőpontokkal lehet fejleszteni. Mivel a játékban exponenciálisan nőnek az értékek ezért lesz olyan segítő, ami növeli a másodpercenként kapott kreditek számát. Végül az arany, ami a legkülönlegesebb fizetőeszköz, lehet vele egyszerűsíteni, felgyorsítani a játékmenetet, több és jobb segítőkhöz és eszközökhöz jutni, mint a másik két fizetőeszközzel.

4 Fejlesztői specifikáció

A szoftverem csomópontokból és jelenetekből áll össze. A csomópont a játék alapvető építőkövei a Godot-ban. A csomópontot gyermekként hozzá lehet adni egy másik csomóponthoz, így egy fát alkotnak. Egy csomópontot példányosítani is lehet, így bármi újra hasznosítható. A jelenet egy kész komponens, amit magában a játékban lehet elhelyezni.

A játék betöltése után a fő jelenet kerül a képernyőre. A jobb felső sarokban a kredit, segítőpont és arany mennyiség található. Ez alatt a legnagyobb tároló található, ami tartalmazza a gyártható termékeket, ez magában egy külön jelenet, öt gombot, ami a termékoldalak közti válogatást viszi végbe és egy gombot, amivel a vásárlási mennyiséget lehet állítani. Alatta pedig az egyéb oldalak között lehet lépegetni.

Négy darab főbb jelenet található a játékban. A nyersanyagokat tároló jelenet, ahol új nyersanyagokat lehet vásárolni és termelni. A második a kreditek fejlesztése oldal, ahol megfelelő nyersanyag mennyiséggel lehet fejleszteni a kredit mennyiségének szerzését. A harmadik jelenet a segítőké, itt lehet megtekinteni őket és itt lehet fejleszteni őket a segítő pontokkal. Végül egy valuta váltó rendszer, ami különböző valutákat enged meg váltani, különböző értékekért.



8. ábra: A játék felépítése

4.1 A nyersanyag tároló jelenet

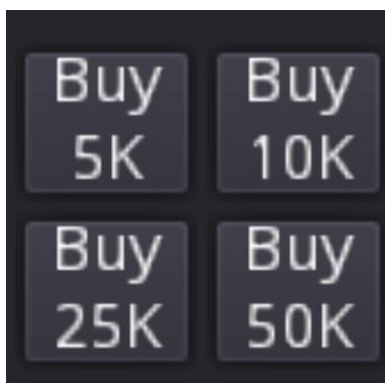
Ez a jelenet önmagában több egybeágyazott jelenetből tevődik össze. Öt oldalon vannak tárolva a nyersanyagok, amit egy jelenet többszöri példányosításával oldottam meg. Mind az ötnek van egy alapanyaga, amit, kredittel együtt új termékeket lehet vásárolni. Egy oldalon belül ismét egy jelenet van többször példányosítva, minden lehetséges információ és felhasználói interakció itt helyezkedik el egy darab nyersanyagról. Lehet vele manuálisan termelni, amit később a segítők automatizálnak, lehet újat vásárolni és vizuálisan meg van jelenítve a termelés folyamata és a nyersanyag mennyisége. Az egy darab nyersanyag jelenete ábra a játékbók kiemelt képen mutatja ezt meg. Ezen kívül elhelyezkedik egy gomb a jelenet jobb felső sarkában, ami megengedi, hogy az elérhető nyersanyagok alapján 10, 25, 50, 100 százalékat költse el az új nyersanyag vásárlására. Az, hogy egy jelenetet többször újra lehet használni, nagyban segíti a kód rövidítését és egyszerűsítését.



9. ábra: Egy darab nyersanyag jelenete

4.2 Kreditek fejlesztése

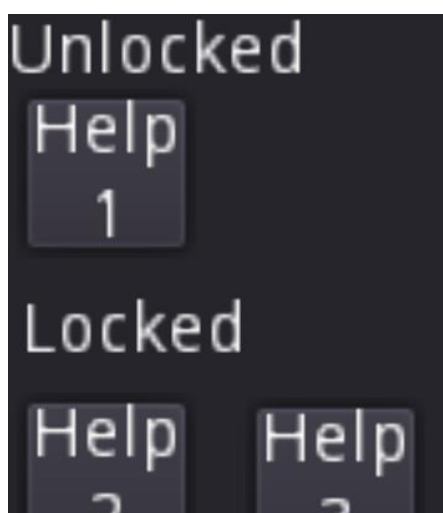
A kreditek fejlesztése jelenet a nyersanyagok oldalát lecserélve egy új oldalra visz minket. Ezen az oldalon öt darab gomb található, mindegyik gomb, ami jelzi, hogy mit tud csinálni. Az első öt gomb a kredit bevétel növelésére van, növeli a minden másodpercen beérkező hozamot. Az első gombbal a jelenetben az első termelői jelenet alap nyersanyagával lehet növelni a kredit termelési mennyiségét. Az ára ennek, először ötezer, majd ötmillió és így tovább az ára. A hatodik gomb egy különleges ajánlat, amit arannyal lehetséges megvásárolni. A kredit fejlesztés jelenet ábrán láthatóak ezek a gombok.



10. ábra: A kredit fejlesztés jelenet

4.3 A segítők

A második fő jelenet a segítőknek ad otthont. Felépítése két részre van osztva. Az egyik részben a feloldott segítők találhatók, míg a másikban a feloldásra várók. A nem elérhető segítők csak képek, nem lehet velük semmit csinálni. Az elérhetők megnyomásával, egy felugró ablak megjeleníti azt a segítőt és annak tulajdonságait. A segítőt ebben az ablakban lehet fejleszteni a segítőpontokkal. A kártya szintjének megfelelő mennyiségű segítőpontért fejleszteni lehet, ezzel javítva a képességét. Sok fajta képessége lehet egy segítőnek. A legalapvetőbb a nyersanyagokat külön- külön automatizáló segítők. Ez a segítő annyit tesz, hogy mostantól nem szükséges annak a nyersanyagnak a manuális termelése. A továbbfejlesztése gyorsítja egy termelői ciklus idejét. Lehetséges még olyan segítőt szerezni, ami egy nyersanyag, vagy egy egész nyersanyagoldalt termelését szorozza meg. Egy segítő le tudja árazni a termékek megvásárlását egy oldalon.



11. ábra: A segítők jelenete

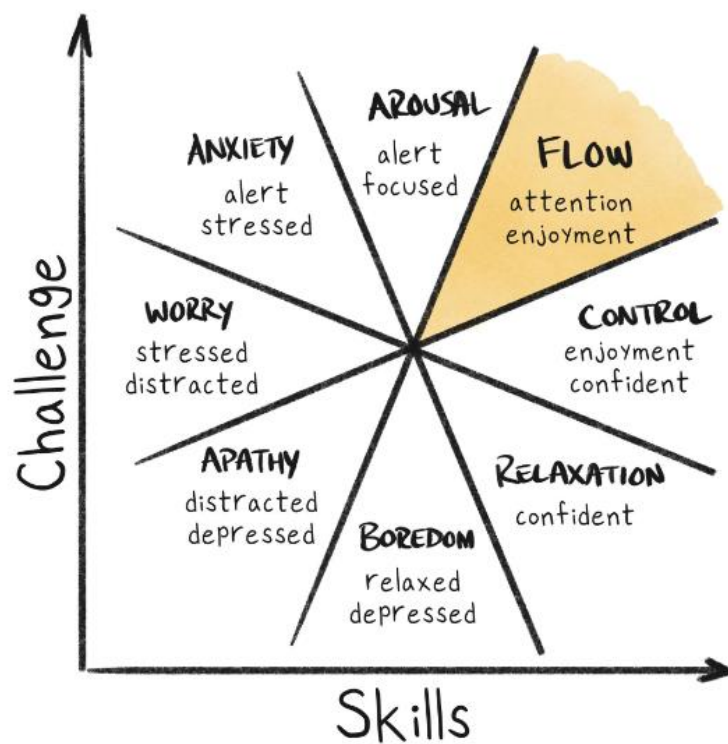
4.5 Valuta váltó

A valuta váltó két részre van osztva. Az első részben kreditet lehet átváltani segítőpontra. Bal oldalon található egy mező ahová a felhasználó beírhatja a mennyiséget, amit szeretne átváltani. Míg a jobb oldalon a gomb, amivel át lehet váltani. Ezek felett található a szöveg, ami jelzi, hogy mit vált és milyen arányban. A második részben kreditet lehet átváltani aranyra. A felépítése ugyan az, mint a segítőpontos váltónak, a felirat kivételével.

4.6 Játékos modellezés

A játékos modellt a játékon belüli adatokra és cselekményekre építem fel. Különböző cselekvés fajtákra fog figyelni a program és ha azok bekövetkeznek, azt feljegyzi magának. Ezek az eseményeket összegyűjtöm, összegzem és megtekintem a gyakoriságát, amiből kikövetkeztetem azt, hogy milyen valószínűséggel fog a játékos bizonyos cselekményeket elkövetni.

Három darab viselkedési formát fogok megtekinteni. Milyen gyakran jelentkezik be a játékba és mennyi időt tölt el benne. Ez azt határozza meg, hogy a játékos mennyire aktív, élvezi-e eléggé a játékot. A második viselkedési forma azt figyeli meg, hogy a játékos milyen eloszlásban foglalkozik az új tartalommal, azaz a prioritása, mint költség és eltöltött idő melyik tartalom irányába húz. A harmadik, amit a program figyelni fog, az a nyersanyagok és pénzek költségének aránya. Egyenletesen költi el a pénzét a lehetőségek között vagy egy valamire fókuszál csak. Ez a második viselkedési formával összekötve megmutatja, hogy a játékos nagyjából hogyan játszik. Ha megfelelően használjuk fel ezeket az adatokat, a kihívás változtatása a megfelelő élményekért ábrán látható függvényben a kijelölt szeletbe tudjuk helyezni a játékélményt.



12. ábra: A kihívás változtatása a megfelelő élményekért

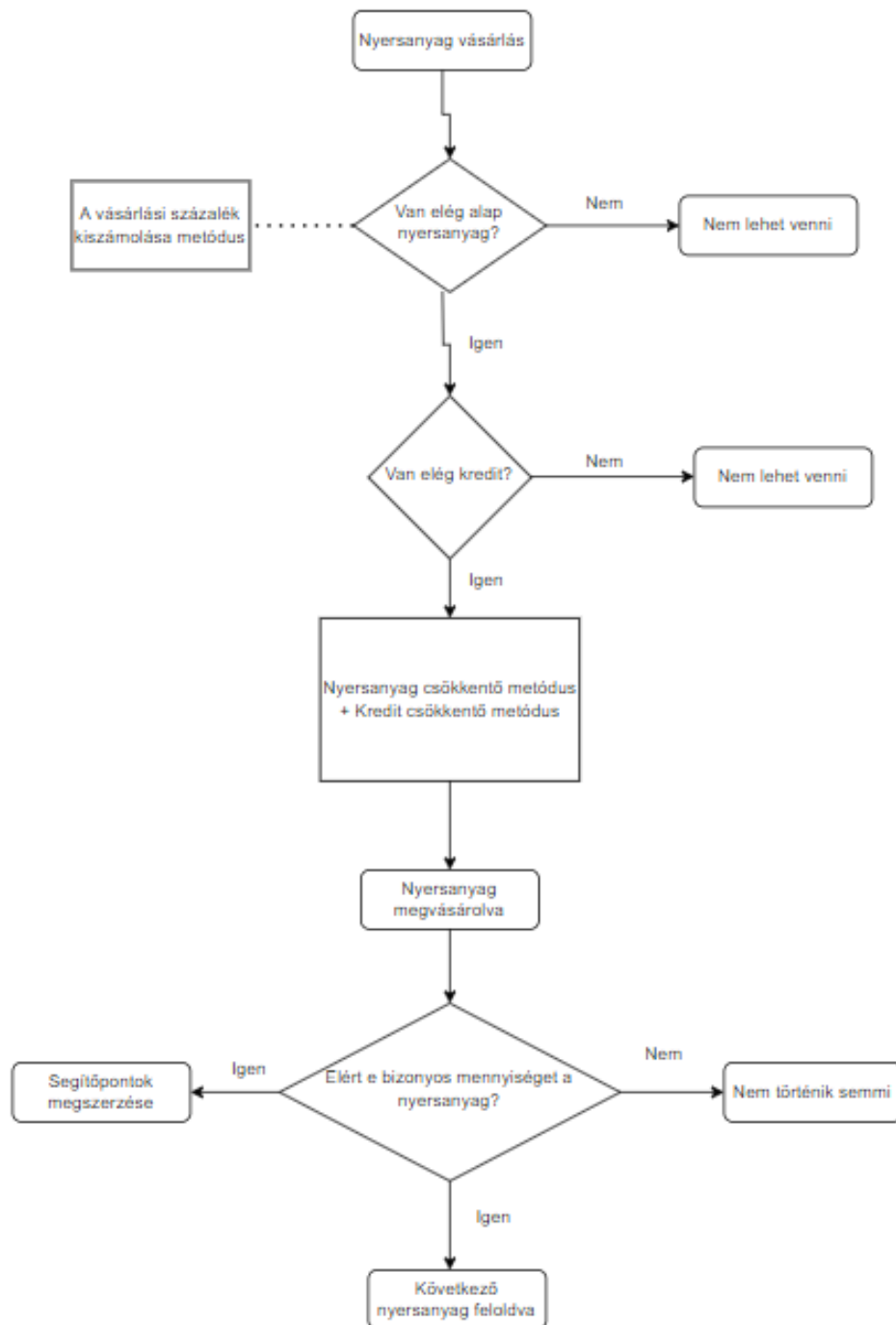
5. A szoftver megvalósítása

Most a program gyakorlati elkészítésének lépéseit írom le. Az irodalomkutatás segített feltárni a játék stílusának tulajdonságait, meglévő programok erősségeit és hibáit. A fejlesztői specifikációval létrehoztam egy elméleti vázlatot, ezzel nagyban megkönnyítettem a problémák lekódolását.

5.1 A nyersanyag tároló megvalósítása

Egy darab nyersanyag jelenete tartalmaz 3 gombot, kettő címkét és egy százalékmutatót. A bal oldali kör alakú gomb a “termel”, azaz annyiival növeli az előtte lévő termék értékét amennyi van belőle. A gomb alján bekeretezve van egy címke, ami jelzi, hogy mennyi termék van. Ez a címke a nyersanyaghoz tartozó változó tulajdonságaiban automatikusan frissítve van. A gomb megnyomásával elindul egy óra, ami egy minden nyersanyagnál más mennyiségű változó által megadott idő alatt végez a termeléssel. Egyszerre csak egy termék csinálhat ilyet. Tőle jobbra van a százalékmutató, ami a termelés idejét mutatja, a _Process függvény segítségével, ami minden képkockában meg van hívva ezzel a folyamatos növekedést biztosítva. Ez alatt található a vásárlás gomb. Alapvetően a gomb megnyomásával egy darabot vásárol a termékből. Ennek az ára egy darab kredit és egy előre meghatározott mennyiség az oldal alap értékéből. Ezek exponenciálisan nőnek, mivel a termelést is sok különböző faktor növeli, így ki kell egyensúlyozni. A gomb megnyomása egy függvényt hív meg ami le ellenőrzi megvannak-e a megfelelő feltételek a vásárláshoz és ha igen megtörténik a vásárlás. A vásárlás mennyiségét lehet változtatni a nyersanyagok és az oldal váltó gombok között elhelyezett gombbal, amire rákattintva változik egy darabról 10%-ra, majd 25%, 50% és végül maximum vásárolható mennyiségre. Egy függvény egy belső időzítővel minden másodpercben kiszámolja, hogy mennyi a maximum, amit lehet vásárolni figyelembe véve a vásárláshoz szükséges összes összetevőt, majd leosztja a jelenleg kiválasztott százalékra. Ezt a vásárlás gombon lévő címke automatikusan kiírja, hogy mennyi. Tőlük jobbra található egy gomb, aminek megnyomásával egy felugró ablakban megnézhetjük azokat a segítőkét, akik ehhez a nyersanyaghoz kapcsolódnak. Amennyiben valamelyik még nincs feloldva csak a fekete körvonalát látjuk, amennyiben meg fel van, tudjuk fejleszteni és megtekinteni a képességeit. Minden nyersanyagból el kell érni egy előre meghatározott mennyiséget, hogy a következőt feloldjuk. Minden oldalon nyolc darab termék található. Az öt oldalból először csak az első elérhető, később oldhatjuk fel a további négyet, különlegesebb termékek gyártására. Ha a feloldás előtt kattintunk arra az oldalra a nyersanyag lista helyett egy üzenet van jelen, ami megmondja mikor férhetünk hozzá. A nyersanyag tároló legbonyolultabb metódusa a termelés, ami a nyersanyag vásárlás menete ábrán található. Először a metódus megtekinti lehetséges-e a

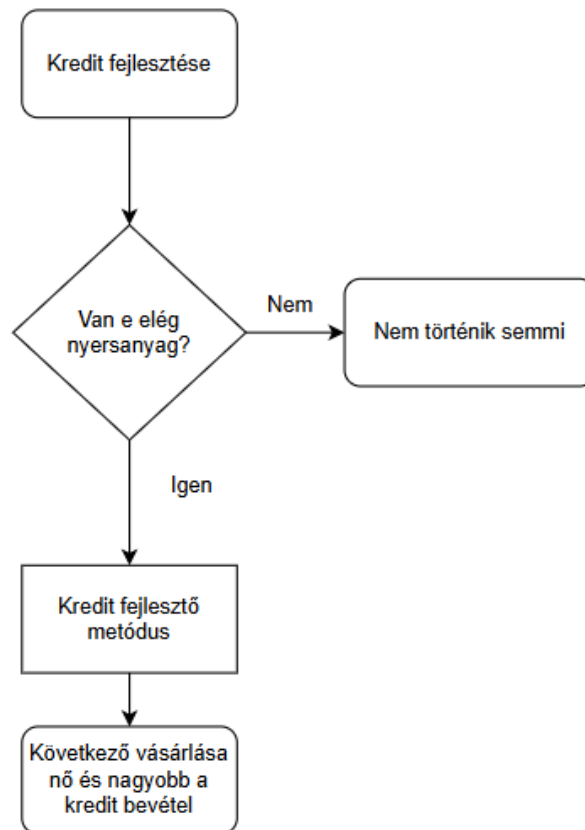
vásárlás úgy, hogy kiszámolja a beállított vásárlás százalékkal azt, hogy mennyi nyersanyag és kredit szükséges hozzá. Ha van, két metódust meghív, ami csökkenti ezeknek az értékét a tárolókban. Ezután le ellenőrzi, hogy elért-e egy bizonyos mennyiséget és ha igen a megfelelő metódust meghívja hozzá.



13. ábra: A nyersanyag vásárlás menete

5.2 A kreditek fejlesztésének megoldása

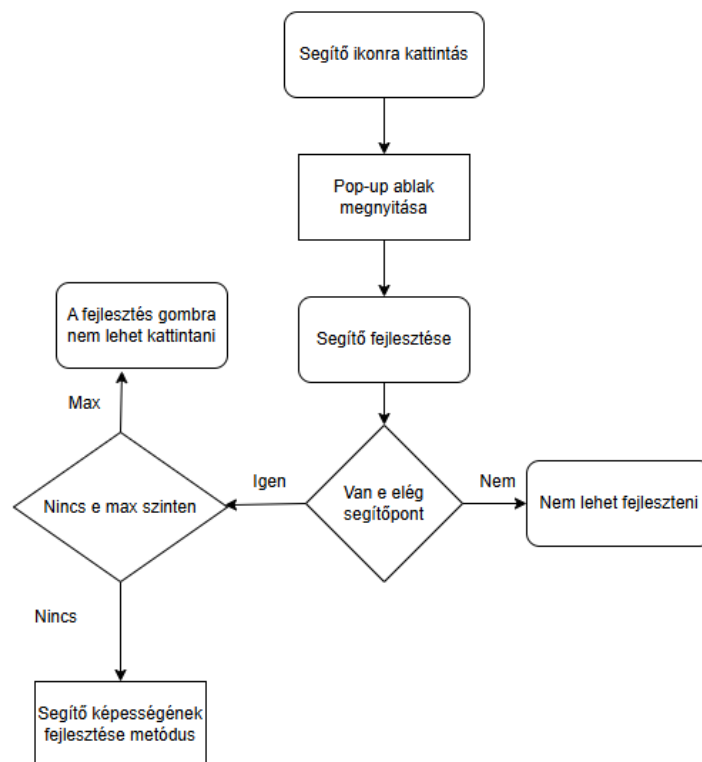
A legalsó vonalon lévő gombok kattintásával lehet jelenetet váltani. Az első gomb megnyomásával, a nyersanyagok jelenete lecserélődik a kreditek fejlesztésének jelenetére. Ha a gombot újra megnyomjuk visszakerülünk a nyersanyagok oldalra. Az oldalon lévő hat gomb mindegyikén van egy címke, ami jelzi, hogy mit tud csinálni és, hogy mennyibe kerül a megvásárlása. Ha a játékosnak van elég alap nyersanyaga, hogy megvásároljon egy kredit hozam fejlesztést, a gomb megnyomásával elindul egy esemény, ahol az ár le lesz vonva és egy számláló megnő egyel. A számláló határozza meg, hogy a következő vásárlás mennyibe kerül egy egyszerű szorzással és ez alapján is kerül ki a gombok címkéjére az ár. Mind az öt nyersanyag oldalnak van egy ilyen számlálója, ami az öt gombhoz tartozik. Ha nincs elég összeg akkor nem csinál semmit.



14. ábra: A kreditek fejlesztésének rendszerterve

5.3 A segítők megvalósítása

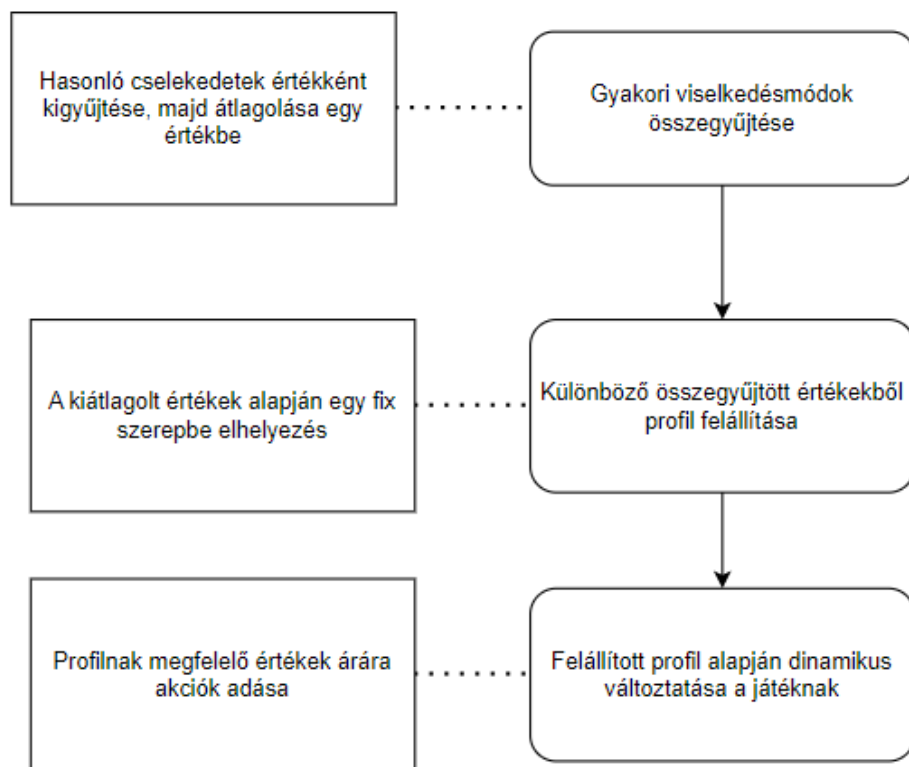
A segítő jelenet felépítése két külön tárolóból áll. A felső tárolóban a már feloldott segítők vannak, míg az alsóban az olyanok, amik még vagy nem elérhetők szint miatt, vagy csak egyszerűen nincsenek még megszerezve. A két tároló dinamikusan méretezi újra magát, tehát ha kikerül egy elem a még nem feloldottak közül és belekerül a másikba, a feloldott tároló automatikusan nő és a párja automatikusan csökken. A nem elérhetőkkal nem lehet semmit csinálni. Az elérhetők pedig egy gomb, aminek megnyomásával egy felugró ablak megjelenít egy képet a segítőről, alatta egy címkét, ami a képességét írja le. Ez a címke a segítő fejlesztése után automatikusan változik. Alatta egy gomb található, amivel lehet fejleszteni a segítőt. A segítő fejlesztéséhez megfelelő mennyiségű segítőpontot kell összegyűjteni és csak akkor lehetséges a fejlesztés gombra kattintani. Ha ez megvan a kártya szintjének megfelelő mennyiségű segítőpontért fejleszteni lehet, ezzel javítva a képességét. Minden segítőt ugyan azzal a pénzzel lehet fejleszteni. Amennyiben egy segítő “életben van”, azaz feloldottuk egy logikai változó átbillen és a szorzója aktiválódik arra a nyersanyagra, amin dolgozik.



15. ábra: A segítők rendszerterve

5.4 A játékos modellezés megvalósítása

A modell összetevőit egy, nem jelenethez kapcsolódó osztályban van elhelyezve. Mind a három egy listához adja a bejövő adatokat és amikor bizonyos számú adat összegyűlt azokat egy értékelés alapján átlagolja és az abból kapott eredményt hozzá adja a végleges számlálókhoz és törli a listák tartalmait. Az átlagolt adat lehet 1, -1 és 0. A végleges számláló így két irányba mozdulhat el, negatív és pozitív. Minél jobban elmozdul az egyik irányba ez a számláló annál jobban egy előre meghatározott viselkedésformát mutat a játékos míg, ha a nullához közel marad a játékos nem koncentrál aktívan azzal a bizonyos viselkedésformával. Végül a három pont összekombinálásával, aminek három "iránya van" kilenc darab különböző játékos típust tudunk meghatározni. A játékos modellezés rendszerterve ábra ezt ábrázolja.

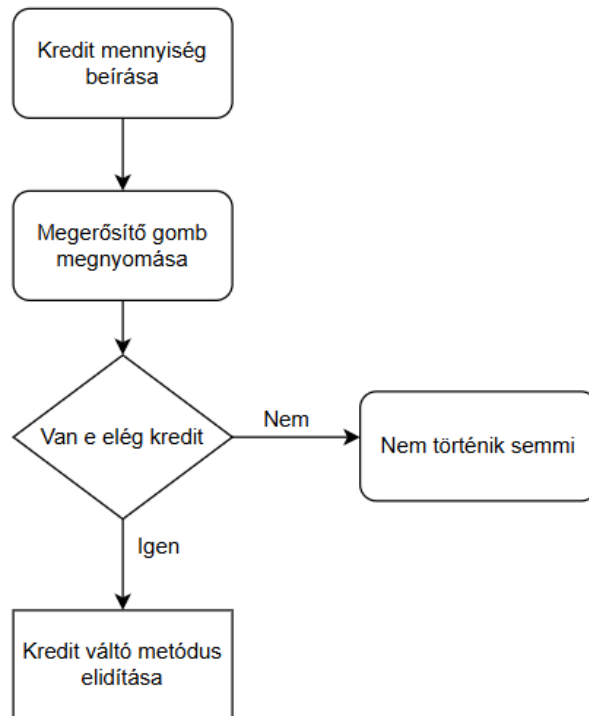


16. ábra: A játékos modellezés rendszerterve

5.5 Valuta váltó

A valuta váltóban két darab váltó található, amik megegyeznek működésben. A tetején található egy szöveg doboz. A szövegdoboz alatt található egy szöveg bemenet. Ettől jobbra található egy gomb, ami elindítja a metódust. A metódus megkapja milyen értéket írt be a

játékos. A metódus először ellenőrzi van e megfelelő mennyiségű kredit az átváltásra. Ha van levonja a kreditből a beírt mennyiséget és hozzáadja az átváltás után a megfelelő mennyiséget a kiválasztott valutához.



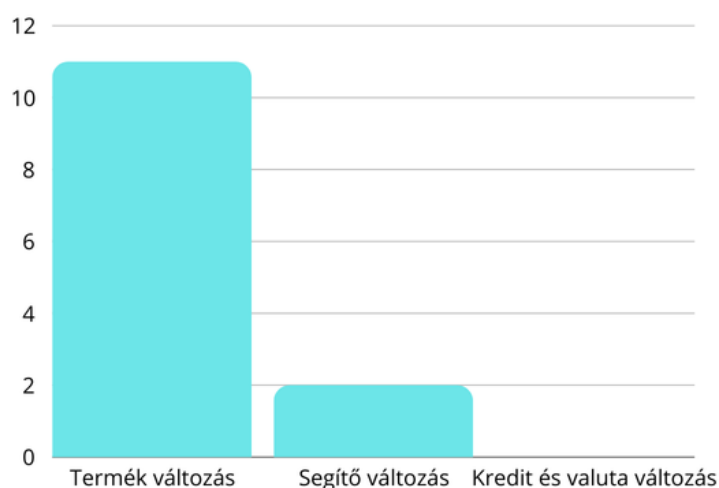
17. ábra: A valuta váltó rendszerterve

6. Eredmények

Ebben a fejezetben az alkalmazás eredményei lesznek kifejtve, a játékos modellezésre koncentrálva. Meg fogom tekinteni, hogy a modellező hogyan reagál a különböző viselkedésformákra és milyen cselekményeket fog végrehajtani válaszként. A teszt úgy lesz végrehajtva, hogy az elkezdett játékban különböző viselkedésformákat hajtok végre és dokumentálom az eredményeket. Három szituációt fogok tesztelni, az elsőben a termékekre fogok koncentrálni és a többi lehetőséggel való interakciót minimalizálom. A másodikban a segítőkre kapják a prioritást míg a harmadik szituációban a kredit fejlesztés és a valuta váltó.

6.1 Termék orientált játék eredménye

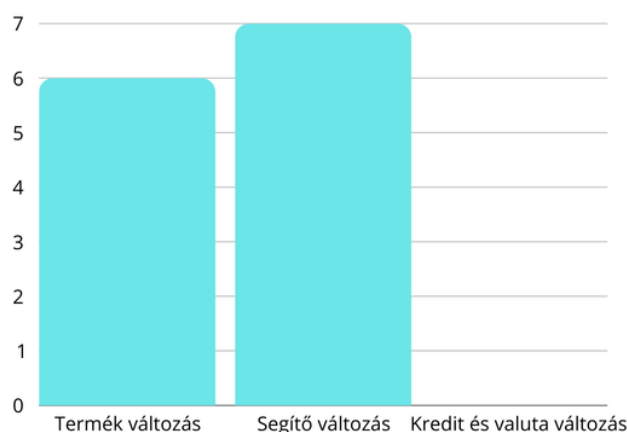
Három darab 30-40 perces játék alatt gyűjtöttem ki az eredményeket. A változás a modellező által felépített profil alapján következik be. A termék orientált játék abban egyedi, hogy itt lehet 100%-ban csak azt fejleszteni, nem szükséges a játék más részével foglalkozni. Két játékban csak a termékekkel foglalkoztam, míg az egyikben 60-40 arányban termék és segítő. Az első kettő játékban mivel csak az eredményekkel foglalkoztam, csak azzal való kedvezményeket adott a modellező, míg a harmadik játékban kétszer a segítőkkal segített és kétszer a termékekkel kapcsolatban. Ha a modell cselekedeteinek számát figyelembe vesszük az arányok megfelelőek.



18. ábra: Termék orientált játék eloszlása

6.2 Segítő orientált játék eredménye

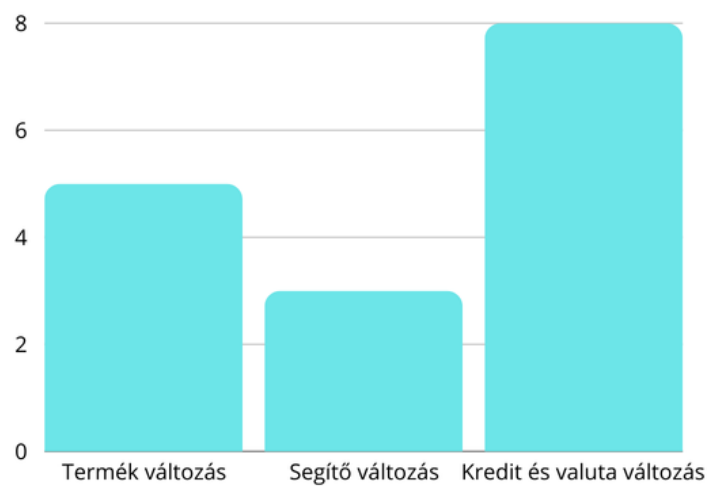
Ismét három játékkal teszteltem az eredményeket. Ebben a szituációban nem lehet csak a segítőkkal foglalkozni, mivel a segítőpontok megszerzése, ezzel a segítők fejlesztése a termékek számának növeléséhez kötött. Így a tesztelés ezen része abból állt, hogy amint van lehetőség segítő fejlesztésére azt rögtön megteszem. Így nagyjából egyenlő számú cselekedetben tudom a segítőket fejleszteni és termékeket vásárolni. A modellező itt is megfelelő módon működik.



19. ábra: Segítő orientált játék eloszlása

6.3 Kredit és valuta orientált játék eredménye

A kredit fejlesztő és valuta váltó elsősorban a termék és segítők fejlesztésének megkönnyítésére szolgál, magában nincs sok értelme használni. Az első körben csak ezt a kettőt használtam, habár a játék kontextusában ennek nincs sok értelme, de a modellező cselekedeteinek ellenőrzésében van. A második kettő játékban kevertem a mindent, de így is a kredit és valután volt a fő hangsúly. Az eredmények itt kicsit jobban szét lettek osztva, de így is megfelelő helyen volt a legtöbb változás az elemző által.



20. ábra: Kredit és valuta orientált játék eloszlása

7. Összefoglalás

A felhasználó modellezés fontos szerepet foglal el az alkalmazások elkészítésében, viszont az elkészítésüknek a bonyolultsága és a köztudat hiánya miatt kevés cég alkalmazza ezt a stratégiát a legnagyobb cégeken kívül. Minden felhasználónak más érdeklődési körei vannak és minél jobban tudja mi az, annál sikeresebb, elégedettséget kiváltó terméket tud nyújtani a felhasználónak a program készítője.

A szakdolgozat célja egy olyan program készítése, ami a játékos viselkedés modellezéssel dinamikusan változtatja egy játék menetét úgy, hogy a játékosnak minél élvezetesebb legyen. A játékos modellezés a felhasználó viselkedéséről, cselekedeteiről gyűjt adatokat, majd előre meghatározott tulajdonságok alapján felépít róla egy profilt. A cél, hogy a felállított profil használatával a játék dinamikusan változzon, a felhasználó által preferált cselekményeket helyezze előtérbe.

A program a Godot Engine nevű játékmotorban készül el, jelenetek és csomópontok összeillesztésével és a mögötte meghúzódó C# nyelven írt logika használatával. A motor a beépített elemeivel segíti a mobil alkalmazás fejlesztését és más motoroktól eltérően rendelkezik dedikált 2D opcióval.

A játékos modellező megbízható módon összegyűjti a játékos cselekedeteit és elmenti a megfelelő kategóriába. Minél tovább játszik a felhasználó annál pontosabb arányban adja vissza azokat a változtatásokat, amik megkönnyítik a játékot, ez által élvezhetőbbé teszik az program használatát.

Összegezve a szakdolgozat betekintést nyújt a játékos modellezés elkészítési és felhasználási módjaira, emellett a játékfejlesztés egyre nagyobb térnyerésére, azáltal, hogy bemutatja az elméleti tervét és gyakorlati megvalósítását egy mobil játékon keresztül. A modellezés minden játékosnak képes személyre szabott, általa preferált élményt nyújtani és értékes betekintést nyújt a felhasználók modellezésének jövőbeli előnyeire és lehetőségeire a játékiparban és azon túl is.

8. Summary

User modeling plays an important role in creating applications, but due to the complexity of their creation and the lack of public awareness, few companies use this strategy except for the largest ones. Every user has different interests, and the better they know what they are, the more successful and satisfying the product the program creator can provide.

The goal of the thesis is to create a program that dynamically changes the course of a game by modeling player behavior so that it is as enjoyable as possible for the player. Player modeling collects data about the user's behavior and actions, and then builds a profile of them based on predefined properties. The goal is to use the established profile to dynamically change the game and prioritize the actions preferred by the user.

The program is created in the Godot Engine game engine, by combining scenes and nodes and using the logic written in the C# language behind it. The engine supports mobile application development with its built-in elements and, unlike other engines, has a dedicated 2D option.

The player modeler reliably collects the player's actions and saves them in the appropriate category. The longer the user plays, the more accurately the changes that make the game easier are returned, thereby making the use of the program more enjoyable.

In summary, the thesis provides insight into the methods of creating and using player modeling, as well as the increasing popularity of game development, by presenting its theoretical design and practical implementation through a mobile game. Modeling can provide each player with a personalized, preferred experience and provides valuable insight into the future benefits and opportunities of user modeling in the gaming industry and beyond.

9. Irodalomjegyzék

- [1] Malone, T.W.: What makes things fun to learn? Heuristics for designing instructional computer games, Palo Alto California, 1980
- [2] Slagle, J.R., Dixon, J.K.: Experiments with the M & N tree-searching program, Commun. ACM 13(3), 1970, 147–154
- [3] Yannakakis, G.N.: How to model and augment player satisfaction: A review, Proceedings of the First Workshop on Child, Computer and Interaction, 2008
- [4] Valencia-Garcia R.: Technologies and Innovation, Second International Conference, 2016, Retrieved 2021-07-22.
- [5] Jesper J.: A casual revolution: Reinventing video games and their players., MIT Press, 2010
- [6] Katie S. and Eric Z.: Rules of Play: Game Design Fundamentals., MIT Press, 2004
- [7] Blair P. and Imran K.: A study of interaction in idle games & perceptions on the definition of a game., IEEE Games Entertainment Media Conference, 2015
- [8] Anthony P.: The Math of Idle Games, Part I, blog.kongregate.com, 2016
- [9] Staffan B. and Jesper J.: Zero-Player Games Or: What We Talk about When We Talk about Players, International Conference on Computer Games, 2012
- [10] Mark E. and Roger E.: Ambient Games, Revealing a Route to a World Where Work is Play?, International Journal of Computer Games Technology, 2008
- [11] Anthony P.: The Math of Idle Games, Part I, blog.kongregate.com, 2016
- [12] Staffan B. and Jesper J.: Zero-Player Games Or: What We Talk about When We Talk about Players, International Conference on Computer Games, 2012

10. Ábrajegyzék

1. ábra: A modellezés folyamata.....	12
2. ábra: A játékos modellezés különböző bemenetei	15
3. ábra: Játéktervezési elvek.....	17
4. ábra: Egy inkrementális játék kinézete	20
5. ábra: Különböző játék motorok előnyei és hátrányai.....	21
6. ábra: Godot Engine jeleneteinek felépítése	23
7. ábra: Exponenciálisan növvő termék ár.....	24
8. ábra: A játék felépítése.....	26
9. ábra: Egy darab nyersanyag jelene	27
10. ábra: A kredit fejlesztés jelenet.....	28
11. ábra: A segítő jelene	28
12. ábra: A kihívás változtatása a megfelelő élményekért.....	30
13. ábra: A nyersanyag vásárlás menete	32
14. ábra: A kreditek fejlesztésének rendszerterve.....	33
15. ábra: A segítő rendszerterve.....	34
16. ábra: A játékos modellezés rendszerterve	35
17. ábra: A valuta váltó rendszerterve.....	36
18. ábra: Termék orientált játék eloszlása.....	37
19. ábra: Segítő orientált játék eloszlása	38
20. ábra: Kredit és valuta orientált játék eloszlása	39