

# GYMNASIUM JANA KEPLERA

Parléřova 2/118, 169 00 Praha 6



## Název maturitní práce

Maturitní práce

Autor: Maximilian Tyx

Třída: 4.B

Školní rok: 2023/2024

Předmět: Informatika

Vedoucí práce: Šimon Schierreich

Praha, 2024





**GYMNASIUM JANA KEPLERA**  
*Kabinet informatiky*

## ZADÁNÍ MATURITNÍ PRÁCE

*Student:* **Maximilian Tyx**

*Třída:* **4.B**

*Školní rok:* **2023/2024**

*Platnost zadání:* **30. 9. 2024**

*Vedoucí práce:* **Šimon Schierreich**

*Název práce:* **Počítačová hra Kramnyk**

*Pokyny pro vypracování:*

Cílem maturitní práce je vytvořit vzdělávací počítačovou hru, která slouží jako názorná ukázka ekonomického mechanismu nabídky a poptávky. Hráč zde soupeří s počítačem ovládaným protivníkem, kterého se snaží porazit v ekonomickém souboji založeném na zjednodušeném modelu trhu.

*Doporučená literatura:*

[1] EVANS, Eric. *Domain-Driven Design: Tacking Complexity in the Heart of Software*. Boston: Addison Wesley Longman Publishing, 2003. ISBN 978-0-321-12521-7.

[2] MARTIN, Robert C. *Design Principles and Design Patterns*. www.objectmentor.com, 2000. Dostupné z: [https://fi.ort.edu.uy/innovaportal/file/2032/1/design\\_principles.pdf](https://fi.ort.edu.uy/innovaportal/file/2032/1/design_principles.pdf).

[3] FOWLER, Martin. *Patterns of enterprise application architecture*. Boston: Addison-Wesley Professional, 2003. ISBN 978-0-321-12742-6.

*URL repozitáře:*

<https://github.com/NexFire/Kramnyk.git>

---

*student*

---

*vedoucí práce*



## Prohlášení

Prohlašuji, že jsem svou práci vypracoval samostatně a použil jsem pouze prameny a literaturu uvedené v seznamu bibliografických záznamů. Nemám žádné námitky proti zpřístupňování této práce v souladu se zákonem č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších předpisů.

V Praze dne 3. března 2024

Maximilian Tyx



## **Poděkování**

Poděkování.



## **Abstrakt**

Tato práce se zabývá prezentací zjednodušeného ekonomického modelu ve formě počítačové hry která je tematicky vložená do středověku

K prezentaci daného ekonomického modelu bylo využito: Game Engine Godot, a programovacího jazyka C#

Po testování a upravování modelu jsem se dostal do momentu kdy modelový protihráči jsou schopni nákupu menšího množství typů položek, které však dle jejich uvážení maximalizují jejich výdělek

Po práci s zjednodušeným modelem ekonomického mechanismu je možné prohlásit že takto zjednodušený model je velmi nepřesný a obsahuje velké množství chyb které znemožňují jeho použití v reálném prostředí. Avšak k použití ve hře je dostačující

## **Klíčová slova**

Godot Game Engine, C#, Ekonomický model, Nabídka,Poptávka

## **Abstract**

This project is supposed to showcase simplified economic model that is presented in a form of PC game

To achieve this i used: Game Engine Godot and a programming language C#

After testing and editing the model. It is capable of purchasing fewer types of product with maximizing its earnings in mind.

After working with this model and testing it i came to conclusion that such a simplified model is not suited for in real life applications, however it is good enough for a game

## **Keywords**

Godot Game Engine, C#, Economic model, Supply, Demand



# Obsah

<b>1 Teoretická část</b>	<b>3</b>
1.1 Důvody pro zjednodušení . . . . .	3
1.2 Rozdílnost modelů . . . . .	3
1.3 Model . . . . .	3
<b>2 Implementace</b>	<b>5</b>
2.1 Využité technologie . . . . .	5
2.2 Problémy s využitými technologiemi . . . . .	5
2.3 Řešení problémů s využitými technologiemi . . . . .	5
<b>3 Technická dokumentace</b>	<b>7</b>
3.1 Instalace . . . . .	7
3.1.1 Spuštění hotového projektu . . . . .	7
3.1.2 Sestavení vlastního projektu . . . . .	7
3.2 Jak Hrát . . . . .	7
3.2.1 Nová hra . . . . .	7
3.2.2 Načtení staré hry . . . . .	10
3.2.3 Uložení hry a odchod z ní . . . . .	10
3.2.4 Ukončení Hry a Zdroje . . . . .	10
<b>Závěr</b>	<b>15</b>
<b>Seznam použité literatury</b>	<b>17</b>
<b>Seznam obrázků</b>	<b>19</b>



# 1. Teoretická část

Tato práce má za cíl navrhnut, vytvořit a implementovat zjednodušený ekonomický model který pracuje na základě ekonomického modelu nabídky a poptávky který byl definován Alfredem Marshalllem, na jehož principu funguje většina dnešního trhu.

## 1.1 Důvody pro zjednodušení

Zjednodušením modelu se má na mysli potřebu zmenšit počet proměnných které ovlivňují trh a chování jednotek nacházející se v něm.

Tato potřeba vzniká z důvodu snazší implementace a zobrazení daného modelu.

## 1.2 Rozdílnost modelů

Kde Marshallův model funguje na principu vyváženosti trhu a zdůrazňuje důležitost obou stran. A popisuje snahy střetnout se v rovnovážném bodu, můj model je spíše zaměřen na nakupování se zaměřením na zisk a model poptávky byl přidán až poté jako dodatečný člen který doplňuje funkčnost mého modelu

## 1.3 Model

Jako nejdůležitější faktory pro prioritu a počet nakoupeného zboží jsem si vybral tyto faktory:

- Průměrná cena za kterou byl produkt již prodán: avgSellPrice
- Průměrná cena za kterou byl produkt již zakoupen: avgBuyPrice
- Počet položek které byly zakoupeny v minulém cyklu (cyklus je definován jako akce koupi zboží, nacenění zboží a prodej zboží): piecesSold

Jako nejdůležitější faktory pro počet položek jsem pak zvolil tyto faktory:

- Velikost produktu (z důvodu uskladnění): spaceComplexity
- Počet položek tohoto typu které již kupující vlastní: countHave
- Počet peněz kterými kupující disponuje: money
- Cena kupovaného zboží: price
- Místo kterým kupující disponuje (z důvodu uskladnění zboží): spaceAvailable

Z těchto faktorů pak vznikne tento vzorec pro počet položek které si kupující bude chtít koupit:

$$\text{početPoložek} = \frac{b \cdot \text{spaceComplexity} \cdot \text{countHave} \cdot (\text{avgSellPrice} - \text{avgBuyPrice}) \cdot \text{piecesSold} \cdot \text{money}}{2 \cdot \text{price} \cdot \text{spaceAvailable}}$$

kde b je koeficient který mění svou hodnotu podle optimalizace modelu v reálné implementaci



# **2. Implementace**

## **2.1 Využité technologie**

K implementaci mnou navrženého modelu jsem využil herní engine Godot. Vedla mě k tomu jeho jednoduchost a schopnost používat k vývoji jazyk C# a modifikovaný jazyk Python (GDScript). Pro generaci pozadí byla využita umělá inteligence specificky DALL-E od OpenAi. A pro některé grafické efekty jsem si stáhnul veřeně dostupné shadery na které odkazují ve hře. Pro některé grafické prvky jsem si stáhnul veřejně dostupné obrázky z ShutterStocku na které také odkazují ve hře.

## **2.2 Problémy s využitými technologiemi**

S oběma jazykama jsem již zkušený a proto byl Engine Godot z počátku jasná volba. To se ukázalo jako lehce problematické protože po té co jsem navrhнул a napsal celý ekonomický model tak spolu GDScript který jsem měl dedikovaný jako jazyk který ovládá grafickou stránku hry a C# ve kterém je napsaný celý mode příliš dobře nekomunikovali, a proto jsem musel celou grafickou stránku hry programovat v C#. Největší problém při implementaci mého modelu bylo zjištění že daný model má mnoho nedostatků v okrajových situacích do kterých se model často dostával, čili bylo třeba tyto okrajové situace vyřešit. Další problém který nastal při práci s grafickou stránkou hry byla nedokončenosť herního enginu Godot kvůli které je spousta funkcí na které jsem zvyklý u UI nefunkčních a nebo vůbec neexistují. Největším problémem Godota se odvozuje z již zmíněné nedokončenosnosti. A to je nedostatek dokumentace a malá komunita uživatelů.

## **2.3 Řešení problémů s využitými technologiemi**

Jako jedno z nejlepších řešení se ukázalo zeptání se komunity na redditu redditu a nebo přímo vyhledat daný problém na Youtubu či Googlu kde některé problémy již dříve řešil. Ačkoliv je jak jsem již zmiňoval komunita Godota dosti malá, odpovědi na můj dotaz se mi dostalo do 12 hodin od položení dotazu

## Implementace

# 3. Technická dokumentace

## 3.1 Instalace

Projekt lze nainstalovat a spustit dvěma způsoby:

### 3.1.1 Spuštění hotového projektu

Je možné spustit již předem sestavený a připravený projekt který se nachází ve Složce:

Export/Release/Kramnyk\_release/Kramnyk\_release.exe

viz obrázek 3.1

### 3.1.2 Sestavení vlastního projektu

Pokud si budete chtít projekt sám sestavit, je to také možné, ale je k tomu potřeba více kroků:

1. Stažení vlastního .NET SDK >=6 zde
2. Stažení vlastního herního enginu Godot zde
3. Exportování vlastního .exe programu
  - Krok 1: 3.2 nalezení export menu
  - Krok 2: 3.3 Výběr cesty k exportu
  - Krok 3: 3.3 Exportování souboru .exe
4. Exportování vlastní složky programu (Krok 4: Obrázek: 3.3)
5. Vložení vlastního .exe programu se vším co se vygenerovalo v kroku 4 do vlastní složky programu

Výsledná struktura složky by měla vypadat jako na obrázku 3.1

## 3.2 Jak Hráť

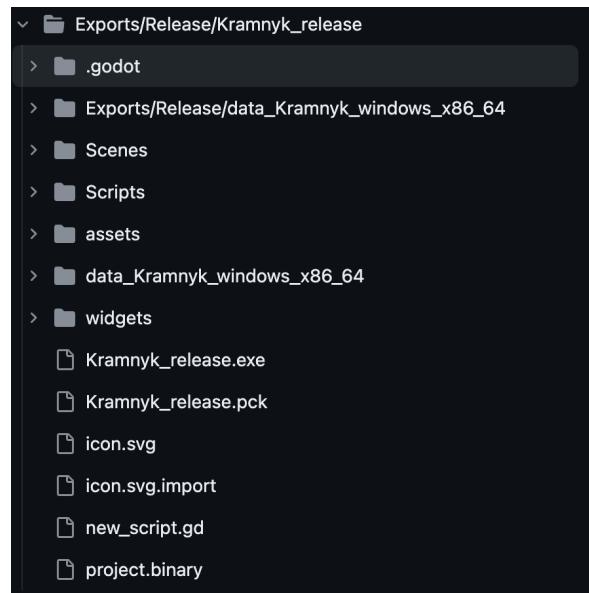
### 3.2.1 Nová hra

3.4

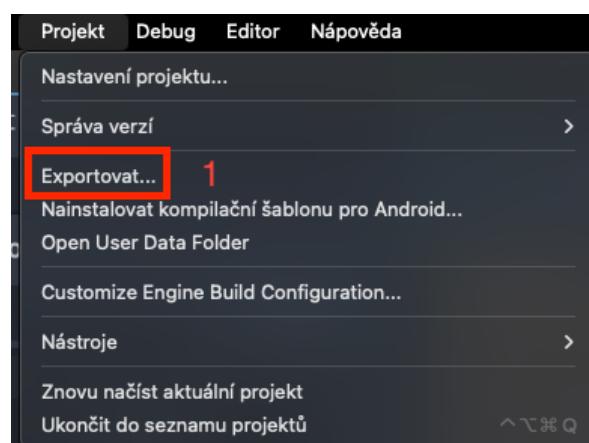
1. Začne novou hru

3.6

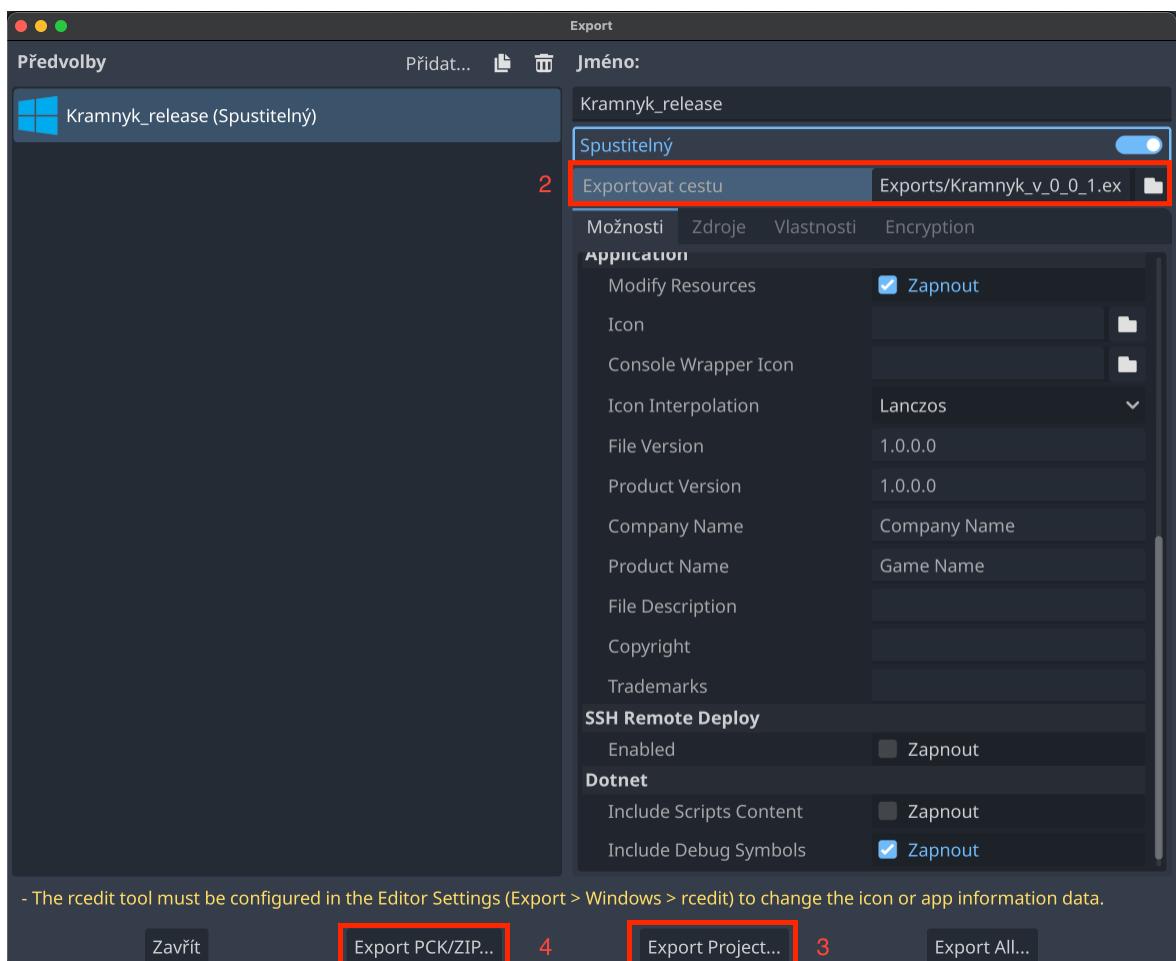
1. Hráč si zvolí za jakou cenu by daný produkt chtěl prodávat
2. Hráč si zvolí množství zboží které chce koupit
3. Hráč zboží koupí
4. Hráč může sledovat kolik mu zbývá peněz



Obrázek 3.1: Zobrazení lokace spustitelného souboru



Obrázek 3.2: zobrazení lokace exportu



Obrázek 3.3: Kroky k exportování projektu

5. Hráč může sledovat kolik místa mu zbývá
6. Pokud je Hráč spokojen s nákupem může pokračovat dál do hry

3.7

1. Zde je vidět Hráčův stav a pokud mu zůstali nějaké itemy tak se zobrazí
2. Pokud hráč chce tak může pokračovat ve hře. Přenesení na 3.6

### **3.2.2 Načtení staré hry**

Pokud už hráč dříve hrál tak se mu načte hra 3.5

1. Hráč si vybere jakou hru chce hrát
2. Tímto jí spustí

### **3.2.3 Uložení hry a odchod z ní**

3.8

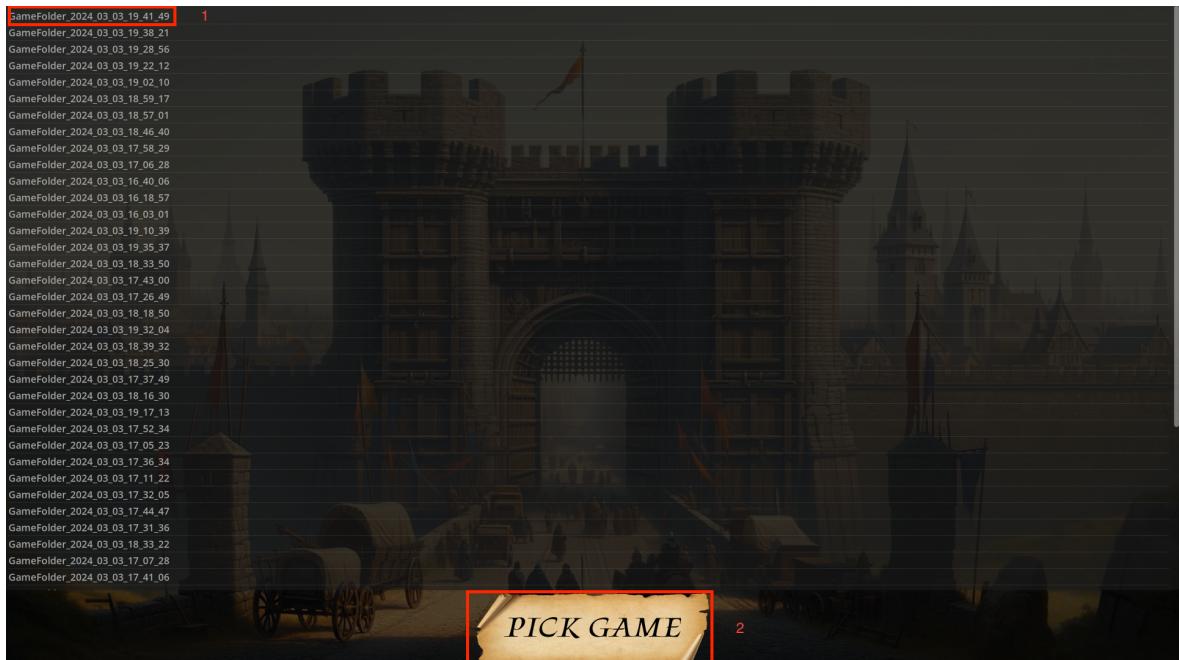
1. Tímto Hráč opustí aktuální hru a ta se zároveň uloží a přenesene hráče na 3.4

### **3.2.4 Ukončení Hry a Zdroje**

3.4 3) Odchod ze hry 10) Odkaz na github kde jsou zdroje použité ve hře zmířené



Obrázek 3.4: Zobrazuje halvní menu hry



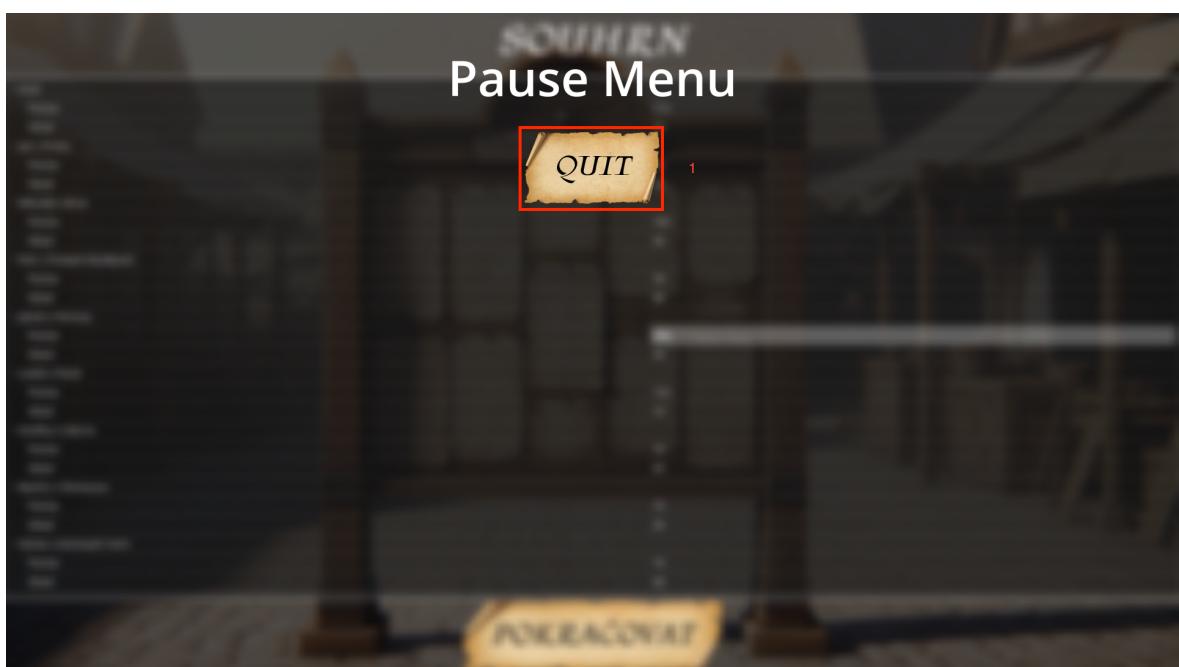
Obrázek 3.5: Zobrazuje menu k načtení staré hry



Obrázek 3.6: Zobrazuje obrazovku k nákupu produktů



Obrázek 3.7: Zobrazuje souhrn hráčů



Obrázek 3.8: PauseScreen



# Závěr

Myslím si že cíl práce byl přibližně splněn. K zjednodušení již dříve zmíněného modelu došlo, ačkoliv se tím jasně prokázalo že takto zjednodušený model je v reálném světě neefektivní a velice nepřesný. Model a protihráči mají tendence se zaměřovat na stejné produkty každý cyklus, a je velice jednoduché je oklamat jakožto hráč. Model by se dal více optimalizovat aby lépe implementoval poptávku. Ale momentální stav je dostačující pro tuto aplikaci

Grafické zpracování hry by mohlo být lepší. Pokud bych věnoval více času studováním jak Godot Game engine funguje, ale myslím si že na prezentaci mého modelu je to dostačující.

Z mého osobního hlediska si myslím že hra dopadla lépe než jsem si myslel že dopadne.



# Seznam použité literatury

- [Hol16] Robert Holman. *Ekonomie*. Praha: C. H. Beck, 2016.
- [JG] Ariel Manzur Juan Linietsky a the Godot community. *Godot Docs – 4.2 branch*. URL: <https://docs.godotengine.org/en/stable/>.



# Seznam obrázků

3.1	Zobrazení lokace spustitelného souboru . . . . .	8
3.2	zobrazení lokace exportu . . . . .	8
3.3	Kroky k exportování projektu . . . . .	9
3.4	Zobrazuje halvní menu hry . . . . .	11
3.5	Zobrazuje menu k načtení staré hry . . . . .	11
3.6	Zobrazuje obrazovku k nákupu produktů . . . . .	12
3.7	Zobrazuje souhrn hráčů . . . . .	12
3.8	PauseScreen . . . . .	13