

**Paweł Kumorowski**

Nr albumu: 378452

***Projekt i wykonanie gry wideo "Treasure Hunter" w silniku Unity***

Praca magisterska na kierunku Informatyka

Praca wykonana pod kierunkiem

dra Marka Badury

w Katedrze Geometrii

Łódź 2023

Spis treści

[Wstęp 3](#_Toc137482956)

[Terminologia 3](#_Toc137482957)

[Zawartość rozdziałów 3](#_Toc137482958)

[Wprowadzenie 5](#_Toc137482959)

[Wykorzystane narzędzia 5](#_Toc137482960)

# Wstęp

## Cel i założenia

Celem pracy było zaprojektowanie oraz wykonanie gry wideo z gatunku roguelite. Głównymi aspektami, na które postawiono nacisk były proceduralne generowanie scenariuszy rozgrywki oraz zróżnicowanie przebiegu kolejnych podejść.

Projekt miał powstać z użyciem darmowych narzędzi oraz assetów dostępnych w Internecie. Miał on zawierać wszystkie funkcjonalności finalnej wersji gry w stopniu co najmniej podstawowym, wraz z gotowym fundamentem pod dalszy rozwój dostępnej zawartości.

W dzisiejszych czasach gatunek gier roguelite jest bardzo popularny. Powstaje wiele tytułów o zróżnicowanej tematyce, stale wnoszących świeże rozwiązania. Niezaprzeczalną zaletą tego gatunku jest możliwość odniesienia komercjalnego sukcesu nawet niewielkim studiom, jak chociażby Massive Monster z grą Cult of the Lamb. Gry tworzone są często z wykorzystaniem już gotowych silników, np. Unity lub Unreal Engine. Są one bardzo rozbudowane oraz stale rozwijane, zawierają więc najnowsze rozwiązania techniczne oraz w dużym stopniu upraszczają oraz przyspieszają proces twórczy.

## Terminologia

* Roguelite – gatunek gier wideo charakteryzujący się m.in. proceduralnym generowaniem świata oraz wysoką regrywalnością.
* Regrywalność – pojęcie określające jak bardzo atrakcyjne dla gracza jest kolejne przejście gry.
* Assety – gotowe elementy pozwalające na szybkie wdrożenie ich do projektu.

## Zawartość rozdziałów

1. Wprowadzenie – dokładne omówienie celów i założeń pracy, wykorzystanych narzędzi oraz przykłady podobnych gier wideo.
2. Dokumentacja programistyczna – opis skryptów i rozwiązań, które są najistotniejsze z perspektywy tematu pracy, ogólna struktura pracy.
3. Dokumentacja użytkownika – instalacja, uruchomienie, poradnik oraz opis rozgrywki.

# Rozdział I. Wprowadzenie

## Wykorzystane narzędzia

Gra „Treasure Hunter” została wykonana w Unity. Silnik ten jest darmowy do celów niekomercyjnych, posiada obszerną dokumentację oraz bazę poradników tworzonych przez użytkowników. Istotnym jego elementem jest również integracja z Unity Asset Store, czyli bazą assetów gotowych do natychmiastowego wykorzystania.

Skrypty w Unity pisane są w języku C#. Do tego celu wykorzystano środowisko Visual Studio, które zawiera szereg ułatwień w ich tworzeniu, m.in. IntelliSense, czyli pomoc w uzupełnianiu kodu w postaci podpowiedzi, debugowanie, czy też możliwość uruchomienia go z poziomu Unity.

Do przechowywania historii oraz kopii zapasowej skorzystano z systemu kontroli wersji GIT w serwisie github.com oraz programu GIT Bash. Ułatwiło to również synchronizację pracy na różnych urządzeniach bez konieczności każdorazowego przenoszenia całego projektu.

## Szersze omówienie projektu

W projekcie zaimplementowano kilka systemów opartych na losowości. Składają się na nie:

* Generowanie labiryntu pomieszczeń przy użyciu algorytmu DFS
* Wybór wnętrza dla każdego z pomieszczeń z zestawu już predefiniowanych
* Pojawianie się przeszkód, dodatkowo modyfikujących wnętrza
* Wybór scenariusza rozgrywki dla każdego pomieszczenia
* Przypisanie pomieszczeniom typów przeciwników i szans na ich pojawienie się

Pełnią one ważną rolę w kwestii regrywalności i dzięki nim każda rozgrywka wygląda inaczej.

Do stworzenia wszelkich elementów dźwiękowych oraz graficznych, tj. modele, animacje, tekstury, miały zostać wykorzystane darmowe, gotowe rozwiązania.