**Specyfikacja wymagań gry komputerowej "WIEŻA".**

Spis treści

Spis definicji

1. Wstęp…………………………………………………….….……5
2. Cel Gry……………………………………………………………5
3. Wymagania środowiskowe……………………………….………5
4. Wymagania funkcjonalne………………………………………...5-6
5. Opis interfejsu użytkownika……………………………………...6-7
6. Wymagania niefunkcjonalne………………………………..……8
7. Instalacja gry………………………………………………..……8
8. Terminarz Tworzenia……………………………………….……9
9. Testy akceptacyjne…………………………………………….…9

Spis rysunków

* Rysunek 1 Poglądowy schemat interfejsu gry
* Rysunek 2 Poglądowy schemat menu
* Rysunek 3 Poglądowy schemat strony tytułowe

**Spis definicji**

* **Strona Tytułowa - Odnosi się do obrazu wyświetlanego bezpośrednio po otwarciu aplikacji**
* **Menu Gry – Interaktywny obraz wyświetlany po wygaszeniu strony tytułowej, zawierający opcje: Nowa gra, Wczytaj poziom, Autorzy, Wyjście poruszamy się po nim strzałkami**
* **Nowa gra -Wyświetla Okno Tekstowe**
* **Okno Tekstowe – Okno wyświetla tekst, fabularną część gry, po zamknięciu wyświetla pierwszy poziom**
* **Poziom – Mapa na której wyświetlone są elementy gry, w fazie początkowej będą dostępne dwa poziomy**
* **Elementy Gry – Bitmapy postaci, przeszkód, wrogów oraz tło poziomów wyświetlane na ekranie**
* **Bitmapa - Pliki graficzne tworzące rozgrywkę**
* **Tło Poziomów - Niekolizyjne bitmapy umieszczone głównie ze względów estetycznych**
* **Bitmapa Postać - Bitmapa którą możemy poruszać w lewo, prawo i w górę w trakcie rozgrywki, ma stałą prędkość**
* **Bitmapa Przeszkoda - Statyczny element gry, bitmapa która w trakcie kolizji z innymi elementami gry odbija je bądź zatrzymuje**
* **Bitmapa Wyjście - Statyczny element gry, bitmapa, która w trakcie kolizji z Bitmapą Postaci wyświetla kod poziomu**
* **Kod Poziomu – Czterocyfrowe hasło wyświetlane po przejściu poziomu, jest przypisane do kolejnego poziomu, po zamknięciu wyświetla kolejny poziom**
* **Bitmapa Wróg Statyczny – Statyczny element gry, bitmapa która w trakcie kolizji z Bitmapą Postaci wyświetla informację o przegranej**
* **Informacja O Przegranej – Okno wyświetlane pod danymi warunkami, pokazuje napis "Przegrałeś" po zamknięciu wyświetla menu**
* **Bitmapa Wróg Dynamiczny - Dynamiczny element gry, bitmapa poruszająca się zgodnie z Kierunkiem Wroga ze stałą prędkością, zaczynająca się poruszać w momencie pojawienia się na Ekranie Gry, przy zderzeniu z Bitmapą Przeszkoda zmienia Kierunek Wroga na przeciwny, w trakcie kolizji z Bitmapą Postaci wyświetla Informacje o przegranej**
* **Ekran Gry - Wyświetla kawałek Poziomu wyśrodkowany na Bitmapę Postaci**
* **Kierunek Wroga - Strona w którą porusza się Wróg Dynamiczny w trakcie zderzenia z Bitmapą Przeszkody zmienia kierunek na przeciwny**
* **Grawitacja – Wektor skierowany w dół zgodnie z którym poruszają się wszystkie bitmapy dynamiczne, dopóki nie trafią na Bitmapę przeszkody lub Dolną Krawędź Ekranu**
* **Dolna Krawędź Ekranu – obszar umiejscowiony w dolnej części wyświetlanej mapy, po zderzeniu z Bitmapa Wróg Dynamiczny, usuwa jego bitmapę, po zderzeniu z Bitmapa Postać wyświetla Informację O Przegranej**
* **Bitmapa Atak Postaci – Akcja Wykonywana przez Postać po wciśnięciu konkretnego przycisku , zastępuje Bitmapę Postaci, Bitmapą Ataku Postaci, po zderzeniu od strony wektora kierunku tej bitmapy z Bitmapa Wróg Statyczny, Bitmapa Wróg Dynamiczny usuwa te bitmapy, trwa nie dłużej niż sekundę**
* **Wczytaj Poziom - Wyświetla okno, na którym można wpisać kod poziomu, po zaakceptowaniu wyświetla poziom do którego jest przypisane**
* **Autorzy - Wyświetla okno z autorami**
* **Wyjście - Kończy pracę programu**
* **Gracz – Osoba która uruchomiła grę**
* **Animacja Końcowa - Animacja wyświetlana po zderzeniu Bitmapy Wyjścia i Bitmapy Postaci na drugim poziomie**

1 Wstęp

Dokument ten przedstawia specyfikację gry Wieża. Gra ta zostanie wykorzystana podczas zajęć Proceduralne Języki Programowania w celu zaliczenia przedmiotu

2 Cel gry

Celem gry jest trenowanie zręczności, wyobraźni oraz rozrywkę.

Ze względu na niski koszt wykonania zostanie ona udostępniona wszystkim chętnym, którzy zgłoszą się do jednego z autorów. Dzięki temu więcej osób będzie miało szansę w nią zagrać.

Grupę docelową użytkowników gry tworzy młodzież powyżej 12 roku życia oraz wszystkie osoby chętne zagrać w tą grę.

3 Wymagania środowiskowe

Gra może zostać napisana w języku C bądź C++ z wykorzystaniem biblioteki allegro. Kod źródłowy ma zostać napisany środowisku Code::Blocks lub Visual Studio.

4 Wymagania Funkcjonalne

Gra Wieża składa się zarówno z 2 części

* Część dla graczy
* Część dla programistów

4.1 Część dla graczy

Jest to aplikacja udostępniona na zasadach licencji WTFPL. Ma ona za zadanie:

* Umożliwić graczom rozpoczęcie gry
* Generować stronę tytułową, menu oraz wszystkie elementy gry
* Umożliwić graczom przejście gry
* Pozwolić graczom na osiągnięcie satysfakcji z przejścia gry

4.2 Część dla programistów

Jest to kod źródłowy udostępniony na zasadach licencji WTFPL. Ma on za zadanie:

* Wyświetlić kod źródłowy oraz wszystkie potrzebne do działania poprawnego pliki
* Poszerzyć wiedzę programistyczną
* Działać jako projekt do tworzenia bardziej zaawansowanych aplikacji bądź gier z wykorzystaniem biblioteki allegro

5 Opis interfejsu użytkownika

Wszystkie pola tekstowe, informacje, nazwy w interfejsie graficznym muszą być w jednym języku, angielskim albo polskim.

5.1 Część dla graczy

Niżej opisane procesy następują jeden po drugim, chyba że zostanie inaczej to opisane

* Po włączeniu programu wyświetla się 3 sekundowa strona tytułowa, jeżeli kliknie się lewy przycisk myszki, od razu się wyłączy, strona tytułowa zostanie dodana w przyszłości przez dział graficzny
* Wyświetlenie Menu z którego można wybrać jedną z 4 opcji, jedna pod drugą w danej kolejności: "Nowa gra", "Wczytaj poziom", "Autorzy", "Wyjście" po lewej stronie aktywnej opcji wyświetla się kółko w kolorze rgb (130, 212, 53),wybór następuje przez kliknięcie spacji, zmiana aktywnej opcji przez strzałkę w górę lub w dół , kolor tła rgb(0,0,0)
* Po wybraniu opcji "Wyjście" zamyka program
* Po wybraniu opcji "Autorzy" wyświetla okno z napisem, który zostanie dodany dodany w przyszłości przez dział human resources
* Po wybraniu opcji "Wczytaj grę" wyświetla okno z napisem "Wpisz kod poziomu", poniżej ma być opcja wpisania czterech cyfr, gdy wpiszemy liczbę 2828 i klikniemy spację to zaczynamy grę od poziomu drugiego przy kliknięciu przycisku ESC wracamy do menu
* Po wybraniu opcji "Nowa gra" włącza nam Okno Tekstowe z opisem fabularnym oraz informacją o sterowaniu, po kliknięciu spacji znika się i pojawia się Pierwszy Poziom gry
  + Okno wyświetla nam Ekran Gry
  + Klikając przycisk ESC przechodzimy do menu
  + Poruszamy postacią Barda, Ekran Gry może przesuwać się jedynie w prawo
  + Warunkiem wygranej poziomu jest dojście Bardem do wyjścia
  + Ruch w lewo, prawo i górę wykonujemy kolejno strzałkami w lewo, prawo, górę
  + Prędkość postaci oraz wrogów jest stała
  + Ruch w górę ograniczony jest 3 wysokościami naszej postaci
  + Spacją wykonujemy atak
  + Na postać oraz wrogów działa grawitacja kierująca ich w dół aż do napotkania przeszkody bądź Dolnej Krawędzi Ekranu
  + W momencie zderzenia z Dolną Krawędzią Ekranu przez postać wyświetla się napis o przegranej i przejście do menu
  + W momencie zderzenia z Dolną Krawędzią Ekranu przez wroga, wróg znika
  + Postać oraz wrogowie nie mogą przechodzić przez przeszkody
  + Przeszkody dzielą się na: podłogę/ścianę/półkę, szafę, lampę
  + W trakcie gry napotkamy wrogów, zderzenie z nimi powoduje wyświetlenie informacji o przegranej i przejście do menu
  + Wrogowie dzielą się na statycznych i dynamicznych, statyczni stoją w miejscu, dynamiczni gdy pojawią się na ekranie zaczynają się poruszać w lewo w momencie zderzenia z przeszkodą zmieniają kierunek ruchu
  + Postać może używać akcji ataku, jeżeli wróg znalazł się w jej zasięgu to znika
  + Wszelkie Bitmapy przeszkód, postaci, ataku postaci i wrogów dział graficzny dostarczy w późniejszym terminie

Okno aplikacji powinno być w rozdzielczości 640x480 i być estetyczne.

* "Rysunek 1 Poglądowy schemat interfejsu gry" oraz "Rysunek 2 Poglądowy schemat menu" -nie odzwierciedla ostatecznego projektu a jedynie ma za zadanie umiejscowienie najważniejszych elementów gry oraz elementów menu.

5.2 Część dla programistów

Dostęp do części dla programistów może być zrealizowany przez kod źródłowy napisany w języku C lub C++ w środowisku Code::Blocks lub Visual Studio.

6 Wymagania niefunkcjonalne

6.1 Wymagania wydajnościowe

* Szybkość renderowania klatek powinna mieć 60 FPS
* Gra powinna ładować się nie dłużej niż 10 sekund
* Wielkość gry nie powinna przekraczać 1GB
* Gra nie powinna mieć błędów
* Gra nie powinna się sama wyłącza

6.2 Wymagania bezpieczeństwa

* Poziom drugi musi być zabezpieczony kodem
* Kod Powinien zawierać 4 cyfry
* Kod nie powinien się zmieniać

6.3 Wymagania niezawodności

* W ciągu roku średnia ilość awarii w jednym dniu nie powinna być większa niż 1
* Akcje po wciśnięciu klawiszy nie powinny być dłuższe niż 1 sekunda

7 Instalacja Gry

Nie jest wymagany instalator gry. Wieża ma być grą przenośną, która może być przechowywana jako aplikacja z kodu źródłowego w pliku .zip

8 Terminarz tworzenia

24.11.17-Zainicjowanie strony tytułowej i Menu

01.12.17-Stworzenie widoku autorów, okna tekstowego i widoku wczytania poziomu

08.12.17- Stworzenie bitmapy 1 i 2 poziomu wraz z elementami gry

15.12.17- Stworzenie zderzeń między obiektami oraz grawitacji

22.12.17- Stworzenie sterowania, ruchu postaci \*Testy

05.01.18- Ruch postaci dynamicznych \*Testy

12. 01.18-Przejście z jednego poziomu do drugiego, wczytywanie 2 poziomu \*Testy

19.01.18- Stworzenie zakończenia gry \*Testy

9 Testy Akceptacyjne

Wykonawca zapewni czas niezbędny na testy i odbiór aplikacji przez Zamawiającego. W przypadku konieczności wprowadzenia poprawek i uzupełnień, czas na ponowny odbiór aplikacji przez Zamawiającego jest do ustalenia.