

# Temat pracy: Stworzenie gry RPG z użyciem grafiki 2D.

Paweł Kamiński

13 października 2015

## 1 Zarys oczekiwanego efektu końcowego

Celem jest stworzenie gry 2D zawierającej mechanikę gry inspirowaną grami z serii Final Fantasy.

## 2 Planowane moduły gry

- mapa świata
- mapy lokalne
- pole bitwy
- ekran menu głównego
- ekran rozwoju postaci
- ekran ekwipunku
- ekran ustawiania opcji
- ekran wczytywania gry
- ekran zapisywania gry

Przełączanie pomiędzy poszczególnymi modułami będzie możliwe dzięki użyciu State Machine.

## 3 Założenia ogólne

Gracz może poruszać się pomiędzy kilkoma lokacjami, wybierając lokację, do której chce się udać, z mapy świata. Po wybraniu danej lokacji gracz może poruszać się po mapie lokalnej. Na mapie lokalnej może dojść do następujących wydarzeń:

- Gracz opuszcza mapę lokalną korzystając ze zdefiniowanego wyjścia. W takim wypadku gracz jest przenoszony na mapę świata.
- Gracz napotyka przeciwnika. W takim wypadku gracz jest przenoszony na pole bitwy.
- Gracz napotyka NPC (inną postać). W takim przypadku:
  - jeżeli jest to postać znacząca, przeprowadza dłuższy dialog z daną postacią, może dostać zadanie lub ukończyć zadanie
  - jeżeli jest to postać nieznacząca, może przeczytać krótkie zdanie wypowiediane przez daną postać, bez wpływu na przebieg fabuły
- Gracz znajduje przedmiot. W takim przypadku przedmiot dodawany jest do ekwipunku.

## 4 Założenia odnośnie zadań w grze

Gra nie będzie zawierała zadań pobocznych. Zadania będą mogły być wykonywane wyłącznie w ustalonej kolejności.

## 5 Założenia odnośnie systemu walki w grze

Planowany system walki to turowy system walki z elementami czasu rzeczywistego. W czasie jednej walki postać sterowana przez gracza mierzy się z pewną liczbą przeciwników (od 1 do 3). Każda postać posiada punkty życia (health-Points). Niektóre postaci mogą posiadać dodatkowo punkty magiczne (magic-Points). Gracz i przeciwnicy mogą wykonywać działania tylko wtedy, gdy mają swoją turę.

W przeciwieństwie do klasycznego turowego systemu walki, tury nie muszą występować na zmianę. Każda z postaci posiada licznik odliczający czas wstecz do możliwości wykonania następnego ruchu (remainingWaitingTime). Informacja o czasie pozostałym do wykonania ataku przez przeciwnika jest jawna dla gracza. Gdy któryś licznik osiągnie 0, dana postać ma prawo wykonać akcję. Niektórzy z napotkanych przeciwników mogą być szybsi lub wolniejsi od gracza w swoich działaniach, tzn. każda postać posiada swój własny współczynnik czasu oczekiwania (waitingTime).

Celem akcji wykonywanej w czasie swojej tury będzie zazwyczaj wykonanie ataku odbierającego punkty życia przeciwnikowi (damage). W zależności od etapu gry, mogą być możliwe różne inne rozwiązania, takie jak przyspieszanie działań własnej postaci, spowalnianie działań przeciwnika, rzucanie czarów defensywnych redukujących przyszłe obrażenia itp. Walka będzie polegać na analizowaniu obecnej sytuacji i wybieraniu jak najskuteczniejszej akcji spośród listy dostępnych akcji.

Każda z akcji posiada pewien czas potrzebny na jej przygotowanie. Jeżeli postać zostanie zaatakowana w trakcie przygotowywania swojego ataku, jej tura

przepada. Zazwyczaj akcje wymagające więcej czasu są potężniejsze, jednak ich wybieranie wiąże się z większym ryzykiem, gdyż przeciwnik może w tym czasie wykonać atak. Z punktu widzenia gracza istotne jest analizowanie współczynnika `remainingWaitingTime` dla każdego z przeciwników i dobieranie dłuższych ataków wtedy, gdy jest jeszcze dużo czasu do tury przeciwnika.

Każda postać posiada współczynnik obrony (`defence`). Jest on wyrażony w skali od 0 do 100 i określa, ile procent obrażeń zadawanych przez przeciwnika jest unikanych. Ilość punktów odbieranych przez atak jest wyrażona wzorem:  $\text{effectiveDamage} = \text{damage} * (100 - \text{defence})$

Jeżeli `effectiveDamage` jest większe niż ilość punktów życia posiadana przez postać, która została zaatakowana, odbierana jest liczba punktów równa ilości punktów życia zaatakowanej postaci, tak aby liczba punktów życia nigdy nie była ujemna.

Po każdej turze następuje sprawdzenie, czy liczba punktów życia gracza lub wszystkich przeciwników wynosi 0. Liczba punktów równa 0 oznacza śmierć postaci.

Jeżeli liczba punktów życia wszystkich przeciwników wyniesie 0, walka jest zakończona sukcesem. Gracz jest przenoszony z powrotem do mapy lokalnej. Jeżeli liczba punktów życia gracza wyniesie 0, walka jest zakończona porażką. Gracz otrzymuje propozycję podjęcia ponownej próby.

## 6 Dźwięk

Gra będzie posiadała ścieżkę dźwiękową odtwarzaną w czasie eksploracji, walk oraz na ekranach menu. W tym celu planuję wykorzystać utwory udostępnione publicznie na wolnych licencjach. Rozważam również wykorzystanie dźwięków związanych z wykonywaniem konkretnych działań w czasie walki lub przy znalezieniu przedmiotu. Wszelkie dialogi w grze będą dostępne wyłącznie w formie tekstowej.

## 7 Rozwój postaci

Wraz z postępami w odkrywaniu fabuły gracz dostanie do dyspozycji nowe umiejętności. Gracz może zobaczyć odblokowane umiejętności na ekranie rozwoju postaci. Rozwój postaci jest liniowy i ściśle powiązany z fabułą. Gracz nie ma możliwości wybierania umiejętności do rozwoju, a ekran rozwoju postaci służy jedynie do sprawdzenia, jakie umiejętności są już odblokowane.

## 8 Platforma docelowa

Gra jest skierowana na komputery osobiste z systemami z rodziny Linux oraz Windows.

## 9 Technologia używana do wykonania projektu

Technologia, którą chciałbym wykorzystać to język Lua (<http://www.lua.org/>) oraz framework LÖVE (<https://love2d.org/>) do języka Lua. Jeśli chodzi o tworzenie map, chciałbym wykorzystać edytor map Tiled (<http://www.mapeditor.org/>).

Dlaczego wybrałem taką technologię? Framework LÖVE jest na wolnej licencji, umożliwiającej wykorzystywanie projektu w dowolnym celu. Kod pisany przy użyciu LÖVE działa na Linuksie, Mac OS X oraz Windows. W porównaniu do np. C++ nie trzeba się zajmować wieloma niskopoziomowymi operacjami. Posiada rozbudowaną i dobrze udokumentowaną bibliotekę HUMP (LÖVE Helper Utilities for Massive Progression), która umożliwia wykonywanie przejść pomiędzy poszczególnymi modułami gry (State Machine). Wykonanie typowych czynności, takich jak np. odtworzenie pliku muzycznego sprowadza się do kilku linijek kodu.

## 10 Opisanie zawartości plików

### 10.1 battle.lua

Plik battle.lua zawiera funkcje:

- battleState:update(dt) - funkcja zajmująca się głównie odliczaniem czasu
- resetCounter(o1, o2, o3) - funkcja przywracająca domyślne wartości czasu oczekiwania na rozpoczęcie tury
- colourIfNeeded(expectedValue) - funkcja zmieniająca kolor czcionki w zależności od tego, czy tekst jest wskazywany przez strzałkę

### 10.2 battleBackend.lua

Plik battleBackend.lua zawiera funkcje:

- fight(o1, o2, o3) - funkcja obsługująca liczniki do rozpoczęcia tury gracza oraz przeciwników
- setArrow() - funkcja zapobiegająca wskazywaniu strzałką na martwych przeciwników
- isDead(o1) - funkcja sprawdzająca, czy ktoś jest martwy
- setInitialValues(tmpValues, initialValues) - funkcja przywracająca domyślne wartości atrybutów postaci

### 10.3 battleBackendMakeAction.lua

Plik battleBackendMakeAction.lua zawiera funkcję:

- makeAction(selectedAction) - funkcja wykonująca akcję o indeksie zawartym w argumencie oraz sprawdzająca, czy walka powinna już zakończyć się zwycięstwem

## 10.4 battleBackendTurns.lua

Plik battleBackendTurns.lua zawiera funkcje:

- playerTurn() - funkcja obsługująca rozpoczęcie się tury gracza
- enemyTurn(o1) - funkcja obsługująca całość przebiegu tury przeciwnika

## 10.5 battleBackendWinning.lua

Plik battleBackendWinning.lua zawiera funkcję:

- winning() - funkcja obsługująca zakończenie walki zwycięstwem gracza

## 10.6 battleControls.lua

Plik battleControls.lua zawiera funkcję:

- battleState:keypressed(key) - funkcja obsługująca sterowanie za pomocą klawiatury w stanie walki

## 10.7 battleControlsMoveArrow.lua

Plik battleControlsMoveArrow.lua zawiera funkcje:

- moveArrowDown() - funkcja ustawiająca odpowiednio współrzędne strzałki wskazującej na przeciwnika w przypadku naciśnięcia klawisza strzałki w dół
- moveArrowUp() - funkcja ustawiająca odpowiednio współrzędne strzałki wskazującej na przeciwnika w przypadku naciśnięcia klawisza strzałki w górę

## 10.8 battleDraw.lua

Plik battleDraw.lua zawiera funkcję:

- battleState:draw() - funkcja odpowiadająca za wyświetlanie klatki obrazu

## 10.9 battleLoading.lua

Plik battleLoading.lua zawiera funkcję:

- battleState:enter() - funkcja odpowiadająca za wczytanie plików graficznych i muzyki

## 10.10 enemy1.lua

Plik enemy1.lua zawiera funkcje:

- E\_attack() - funkcja odpowiadająca za wykonanie zwykłego ataku przez Dorvera
- E\_magicAttack(attackingEnemy) - funkcja odpowiadająca za wykonanie magicznego ataku przez Dorvera
- E\_heal(castingEnemy) - funkcja odpowiadająca za uleczenie się Dorvera
- E\_defend(castingEnemy) - funkcja odpowiadająca za bronienie się Dorvera
- Enemy1AI() - niezaimplementowana, docelowo ma zawierać algorytm działań w czasie tury Dorvera, sztuczną inteligencję przeciwnika

### 10.11 main.lua

Plik main.lua to główny plik gry. Zawiera funkcje:

- love.load() - funkcja wczytująca ustawienia, takie jak czcionka, domyślna pozycja gracza itp.

- love.keypressed(key) - funkcja zawierająca obsługę poleceń klawiaturowych działających globalnie w całej grze

- love.draw() - funkcja wywoływana przed wyświetleniem każdej klatki, dostosowuje wyświetlany obraz do rozdzielczości ekranu

### 10.12 map.lua

Plik map.lua zawiera funkcje:

- mapState:update(dt) - funkcja zajmująca się uaktualnianiem mapy i w razie potrzeby przełączeniem w stan walki

- mapState:draw() - funkcja wyświetlająca mapę

- mapState:keypressed(key) - funkcja obsługująca sterowanie w stanie chodzenia po mapie

### 10.13 mapEnterState.lua

Plik mapEnterState.lua zawiera funkcję:

- mapState:enter() - funkcja odpowiadająca za wczytanie plików graficznych do wyświetlenia na mapie

### 10.14 player.lua

Plik player.lua zawiera funkcje:

- attack(o1) - funkcja odpowiadająca za wykonanie przez gracza zwykłego ataku

- magicAttack(o1) - funkcja odpowiadająca za wykonanie przez gracza magicznego ataku

- lockedAction(o1) - ta funkcja nic nie robi. Jest obecna w celach testowych.

- heal() - funkcja odpowiadająca za rzucanie czaru uleczonego

- defend() - funkcja odpowiadająca za akcję obronną

- listOfAllActions() - funkcja odpowiadająca za stworzenie tabeli zawierającej wszystkie dostępne akcje (tzn. z pominięciem tych, które są zablokowane)

### 10.15 playerActionFlags.lua

Plik playerActionFlags.lua nie zawiera żadnych funkcji. Zawarta jest tu tabela zawierająca nazwy akcji, informacje o odblokowaniu akcji, id, opis, liczba potrzebnych punktów magicznych do wykonania oraz czy można obecnie wykonać akcję.

## 11 Planowana struktura katalogów

- mainMenu
- maps: worldMap, levels: level1, level2, level3
- music
- enemies
- battle: animations, battle, draw, player, playerActionFlags, backend: backend, makeAction, turns, winning, controls: controls, moveArrow, loading: animationLayer, backgroundLayer, loading, spriteLayer

## 12 Założenia dotyczące fabuły

Główna bohaterka ma na imię Julia. Każdy poziom składa się z 3-5 zadań.

### 12.1 Etapy projektowania fabuły

- wykreowanie bohatera
- opracowanie typowego dnia z życia tego bohatera
- stworzenie planu ramowego fabuły
- stworzenie planu szczegółowego fabuły

## **12.2 Plan ramowy fabuły**

- 12.2.1 Jest sobie wioska, jakieś wprowadzające zadanie na początek**
- 12.2.2 Zawiązanie akcji, ludzie chorują, opowiedzenie legendy o Neilu Merzentanarze**
- 12.2.3 Julia zdobywa Artefakt Ziemi**
- 12.2.4 Julia zdobywa Artefakt Powietrza**
- 12.2.5 Okazuje się, że należy zebrać cztery magiczne artefakty: Artefakt Ziemi, Artefakt Powietrza, Artefakt Lodu, Artefakt Ognia**
- 12.2.6 Julia zdobywa Artefakt Lodu**
- 12.2.7 Julia zdobywa Artefakt Ognia**

## **13 Fabuła zadań w grze**

### **13.1 Level 1 - miejsce: wioska Lipinki**

#### **13.1.1 Zadanie 1 - Przyjacielska pomoc**

Cel zadania: Usunąć szkodniki z ogrodu Afedii

Początek zadania: Afedia prosi Julię o pozbycie się szkodników jej ogrodu.

Przebieg zadania: Aby ukończyć zadanie, należy pokonać wszystkie szkodniki z ogrodu (prosta walka) i przekazać informację o ukończeniu zadania do Afedii.

Koniec zadania: Po zameldowaniu zabicia szkodników u Afedii.

#### **13.1.2 Zadanie 2 - Lekarstwo dla Osigolda**

Cel zadania: Przynieść lekarstwo Osigoldowi

Początek zadania: Po rozpoczęciu rozmowy z Osigoldem. Do rozpoczęcia zadania niezbędne jest ukończenie zadania przyjacielska pomoc.

Przebieg zadania: Osigold jest poważnie chory. Opowiada historię o magicznej roślinie, kwitaryksonie, która leczy wszystkie ludzkie choroby. Należy pokonać Stwory (nieco trudniejsza walka) i przynieść lekarstwo.

Koniec zadania: Po przekazaniu lekarstwa.

#### **13.1.3 Zadanie 3 - Tajemnicza epidemia**

Początek zadania: Osigold spożywa roślinę, ale nic się dzieje. Co gorsze, podobne problemy dotknęły jego żonę Deatlanę. Osigold opowiada legendę o czarnoksiężniku Neilu Merzenantarze, który kiedyś zaprowadził terror nad światem, a potem został wygnany w zaświaty. Według legendy, gdy Neil Merzenantar powróci z zaświatów, Kwitarykson przestanie uleczать ludzkie choroby, a ludzie mieszkający w wiosce, jedna osoba po drugiej, zaczną chorować. Jeśli



legenda jest prawdziwa, nie pozostało wiele czasu. Afedia może znać sposób na rozwiązanie tych problemów.

Przebieg zadania: Należy porozmawiać z Afedią, która opowie o istniejącym w jej rodzinie magicznym Artefakcie Ziemi, który był przekazywany w jej rodzinie i według relacji babci Afedii miał mieć związek z tajemniczą legendą o Neilu Merzenantarze. Afedia przekazuje swój artefakt i radzi, aby udać się z nim do wiedzmy Tinaldiny mieszkającej w Mrocznym Lesie, która może wiedzieć coś więcej.

Koniec zadania: Natychmiast po rozmowie z Afedią.

## **13.2 Level 2 - miejsce: Mroczny Las**

### **13.2.1 Zadanie 4 - Rozmowa z Tinaldiną**

Cel zadania: Należy porozmawiać z Tinaldiną

Początek zadania: Po wejściu do Mrocznego Lasu

Przebieg zadania: Należy pokonać potwory zagrażające drogę i porozmawiać z Tinaldiną. Tinaldina powie o swoich podejrzeniach, że Artefakt Ziemi może mieć z tą sprawą coś wspólnego. Mówi też, że prawdopodobnie istnieją jeszcze inne podobne artefakty. Ich odnalezienie może być kluczem do ocalenia wioski. W czasie rozmowy zjawia się stwór. Tinaldina prosi o zabicie potwora. Po zabiciu potwora Tinaldina informuje o istnieniu zwoju z przepowiednią, który powinien zawierać dalsze wskazówki.

Koniec zadania: Po rozmowie z Tinaldiną

### **13.2.2 Zadanie 5 - Zwój z przepowiednią**

Cel zadania: Znaleźć zwój z przepowiednią

Początek zadania: Po rozmowie z Tinaldiną

Przebieg zadania: Należy odnaleźć zwój z przepowiednią napisaną w starożytnym języku i zanieść Tinaldinie

Zakończenie zadania: Po znalezieniu zwoju i rozmowie z Tinaldiną

### **13.2.3 Zadanie 6 - Strażnicy Powietrza**

Cel zadania: Zdobyć Artefakt Powietrza

Początek zadania: Po ukończeniu zadania „Zwój z przepowiednią”

Przebieg zadania: Należy pokonać strażników powietrza i zdobyć Artefakt Powietrza

Zakończenie zadania: Po rozmowie z Tinaldiną

## **13.3 Level 3 - miejsce: Kraina Równowagi**

### **13.3.1 Zadanie 7 - Rozmowa z Lidenonem**

Cel zadania: Uzyskać wskazówki dotyczące taktyki od Lidenona

Początek zadania: Po ukończeniu zadania „Strażnicy Powietrza”

Przebieg zadania: Lidenon przekazuje wskazówki dotyczące walk ze strażnikami ognia i lodu

Zakończenie zadania: Po rozmowie z Lidenonem

### **13.3.2 Zadanie 8 - Strażnicy Ognia**

Cel zadania: Zdobyć Artefakt Ognia

Początek zadania: Po ukończeniu zadania „Rozmowa z Lidenonem”

Przebieg zadania: Należy pokonać strażników ognia i zdobyć Artefakt Ognia

Zakończenie zadania: Po rozmowie z Lidenonem

### **13.3.3 Zadanie 9 - Strażnicy Lodu**

Cel zadania: Zdobyć Artefakt Lodu

Początek zadania: Po ukończeniu zadania „Strażnicy Ognia”

Przebieg zadania: Należy pokonać strażników lodu i zdobyć Artefakt Lodu

Zakończenie zadania: Po rozmowie z Lidenonem

## **13.4 Boss**

### **13.4.1 Faza 1**

W miarę standardowa walka, tylko ze zwiększonym poziomem trudności.

### **13.4.2 Faza 2**

Przeciwnik raz na jakiś czas zadaje potężne ciosy pozostawiające tylko 1 HP, po czym daje graczowi szansę na zregenerowanie się.

### **13.4.3 Faza 3**

Julia jest bliska śmierci, ale ratuje ją Afedia.

## **13.5 Zakończenie i napisy końcowe**

# **14 Ścieżka dźwiękowa**

Potrzebuję plików:

- do chodzenia po mapach: wioska Lipinki (już jest), Mroczny Las, Kraina Równowagi
- do walki z każdym z przeciwników: Szkodnik, Stwór (Dorver) (już jest), Potwory zagrażające drodze do Tinaldiny (3 rodzaje), Potwór atakujący w czasie rozmowy z Tinaldina, Strażnicy Powietrza, Strażnicy Ognia, Strażnicy Lodu, Neil Merzentanar - ogółem 10 plików
- menu

- napisy końcowe
- zwycięstwo (już jest)
- porażka (już jest)

Menu: El Postre.ogg, CC-BY 3.0 by Patrick de Arteaga

<http://opengameart.org/content/mml-el-postre>

Napisy końcowe: Miami-Soul, CC-BY-SA 3.0 by neocrey

<http://opengameart.org/content/neocrey-miami-soul>

Mroczny las: Bury My Visions - Flicker, CC-BY-SA 3.0 by Trevor Lentz

<http://opengameart.org/content/night>

Kraina równowagi: Bury My Visions - Unleashed, CC-BY-SA 3.0 by Trevor

Lentz

<http://opengameart.org/content/night>

WALKI:

Szkodnik: oranges-melodies, CC-BY 3.0 by Snabisch

Potwór1: only 115 NV.mp3, CC-BY 3.0 by Snabisch

<http://opengameart.org/content/oranges-melodies>

<http://opengameart.org/content/only-115-new-version>

<https://itunes.apple.com/es/artist/snabisch/id736773346> <https://play.google.com/store/music/artist/S>

[http://www.amazon.com/s/ref=nb\\_sb\\_noss?url=search-alias%3Daps&field-keywords=snabisch](http://www.amazon.com/s/ref=nb_sb_noss?url=search-alias%3Daps&field-keywords=snabisch)

Potwór2: Flaming Soul.mp3, CC-BY 3.0 by Marcelo Fernandez

Potwór3: Battle of the Void.mp3, CC-BY 3.0 by Marcelo Fernandez

Strażnicy powietrza: Small Epic.mp3 by Marcelo Fernandez

Strażnicy lodu: metal ages.wav by Marcelo Fernandez

<http://opengameart.org/content/flaming-soul>

<http://marcelofernandez.tk/>

<https://marcelofernandez.bandcamp.com/>

<https://soundcloud.com/marcelofernandezmusic>

<https://www.facebook.com/marcelofernandezmusic>

Potwór atakujący w czasie rozmowy z Tinaldiną: S31-Through the Portal.ogg, CC0 by section31

Strażnicy ognia: S31-Unexpected Trouble.ogg, CC0 by section31

<http://opengameart.org/content/through-the-portal>

## 15 Przedmioty do używania przez gracza

- mikstura lecznicza (przywraca niewielką część utraconych punktów życia)  
- na każdej z map (mapy 1 - 3)
- eliksir leczniczy (przywraca całość utraconych punktów życia, przedmiot rzadki) - w Mrocznym Lesie (mapa 2)
- antidotum (usuwa efekty trucizny rzuconej przez wroga, trucizna będzie zabierać co turę 1/5 maksymalnych punktów życia) - w Krainie Równowagi (mapa 3)

- trucizna (zabiera co turę 1/5 maksymalnych punktów życia wrogowi) - w Wiosce Lipinki (mapa 1)
- eliksir magiczny (przywraca całość wykorzystanych punktów magicznych, przedmiot rzadki) - w Krainie Równowagi (mapa 3)

Lista przedmiotów do używania w trakcie walki na każdej z map:

- mapa 1: mikstura lecznicza, trucizna
- mapa 2: mikstura lecznicza, eliksir leczniczy
- mapa 3: mikstura lecznicza, antidotum, eliksir magiczny

## 16 Lista przeciwników w zależności od mapy

- mapa 1: thornbush, creature
- mapa 2: forest guardian, cactus, forest creature, dragon, air guardian
- mapa 3: fire guardian, ice guardian
- mapa 4: Neil Merzenantar