Wieża Icy

Dokumentacja

Autorzy:

Mateusz Stanaszek

Mateusz Ścirka

# Opis gry

Fabuła gry wprowadza postać Icy, księżniczki zamkniętej, cytując klasyka „w najwyższej komnacie w najwyższej wieży” i czekającej na księcia z bajki. Kiedy książę jednak nie pojawia się przez dłuższy czas, Icy postanawia wziąć sprawy w swoje ręce. Szybko wydostaje się z komnaty, lecz przed nią droga przez wiele pięter, wypełnionych strażnikami i najeżonych pułapkami.

# Mechanika gry

„Wieża Icy” jest grą platformową z częściowymi zapożyczeniami mechaniki z gatunku tzw. Metroidvanii.

Zadaniem gracza jest ucieczka z wieży, przy uzyskaniu możliwie jak największej ilości punktów. Punkty przyznawane są za rozmieszczone gęsto na planszach monety.

Metroidvania (platformowa gra akcji) jest gatunkiem zapoczątkowanym w 1986 roku przez firmy Nintendo oraz Konami odpowiednio w grach Metroid i Castlevania. Na pierwszy rzut oka są to dwuwymiarowe platformery, jednak wchodząc w szczegóły można wyróżnić trzy dodatkowe mechaniki:

* Labiryntowy rozkład poziomów, do celu nie prowadziła prosta ścieżka, na planszy można było łatwo się zgubić.
* Możliwość wielokrotnego przechodzenia wszystkich poziomów, w dwóch wariantach. W pierwszym przejścia między poszczególnymi poziomami są odblokowywane w czasie gry i nie zostają zamknięte, w drugim dostępna jest mapa z wyborem każdego ukończonego poziomu.
* Rozwój postaci, pochodzący z gier cRPG, objawiający się najczęściej pod postacią:
  + ulepszania posiadanych umiejętności (np. dłuższy skok, większy pasek życia);
  + otrzymywania nowych ruchów (podwójny skok, odbijanie się od ścian);
  + lub otrzymywania nowych broni (broń dystansowa, łańcuch pozwalający dostać się w niedostępne wcześniej miejsca)

# Wykorzystane technologie

Do implementacji projektu wykorzystano język HTML w wersji 5 oraz język JavaScript wraz z bibliotekami jQuery oraz dedykowaną biblioteką do gier – Quintus.

# Opis silnika Quintus

Biblioteka Quintus to oparty o HTML5 oraz JavaScript, zaprojektowany jako modularny i lekki, silnik służący do programowania gier przeglądarkowych oraz mobilnych. Quintus charakteryzuje się prostotą i łatwością używania będąc jednocześnie bardzo elastycznym i wszechstronnym narzędziem, za pomocą którego można stworzyć bardzo rozbudowane gry. Powstał przy tworzeniu książki „[*Professional HTML5 Mobile Game Development*](http://www.amazon.com/gp/product/B0094P2TU6/ref=as_li_ss_tl?ie=UTF8&camp=1789&creative=390957&creativeASIN=B0094P2TU6&linkCode=as2&tag=tun02-20)” i został wyodrębniony do ogólnodostępnego repozytorium. Silnik składa się z następujących, w większości niezależnych od siebie modułów:

* *quintus.js* – główny plik silnika (pętla gry, obsługa klas, wczytywanie zewnętrznych komponentów),
* *quintus\_2D.js* – obsługa grafiki 2D,
* *quintus\_anim.js* – obsługa animacji ,
* *quintus\_audio.js* – obsługa dźwięku,
* *quintus\_input.js* – obsługa wejścia (klawiatura, mysz),
* *quintus\_scenes.js* – obsługa scen (poziomów, kolizji itp.),
* *quintus\_sprites.js* – obsługa obiektów użytkownika (elementy gry, gracze, przeciwnicy itp.),
* *quintus\_tmx.js* – obsługa formatu .tmx, w którym mogą być zapisywane mapy dla gry,
* *quintus\_touch.js* – obsługa dotyku,
* *quintus\_ui.js* – obsługa interfejsu (przyciski, tekst itp.)

Quintus pozwala na tworzenie gier wykorzystując elementy programowania zorientowanego obiektowo, takie jak klasy i obiekty. Wykorzystując mechanizmy JavaScript możliwe jest także stworzenie prymitywnego mechanizmu dziedziczenia. Quintus jednakże wciąż jest we wczesnej fazie rozwoju i podlega ciągłym usprawnieniom.

# Środowisko i narzędzia:

* Notepad++ v.6.7 – rozbudowany edytor tekstu (środowisko deweloperskie)
* Git – system kontroli wersji
* jQuery 2.1.3
* Quintus 0.2.0

Jako platform testowe służą przeglądarki Chrome oraz Firefox w najnowszej dostępnej wersji oraz tablet działające pod systemami iOS oraz Android.

# Pliki projektu:

css/ - katalog ze stylami

data/ - pliki z mapami w formacie TMX

docs/ - dokumentacja

images/ - folder z grafikami

js/ - folder z silnikiem gry

objects/ - klasy obiektów

scenes/ - klasy scen i plansz

index.htm - główny plik wykonywalny

**7. Klasy:**

**Endgame** – jest to klasa, która odpowiada za wyświetlenie sceny podsumowującej rozgrywkę, zliczanie końcowych punktów oraz ich wyświetlanie

**Map, Lvl1, Lvl2** – klasy odpowiedzialne za wyświetlanie plansz

**Menu** – klasa menu

**Skills** – klasa odpowiedzialna za wyświetlanie dodatkowych umiejętności i bonusów, które mogą zostać dodane do rozgrywki za odpowiednią ilość punktów.

**Stats** – klasa odpowiada za wyświetlenie statystyk (ilość życia, ilość punktów)

**Enemy –** klasa odpowiada za zachowania i animację przeciwników

**Player**  - klasa odpowiada za sterowanie, zachowanie i animację gracza

**PortalLvl –** obiekt tej klasy przenosi gracza do następnego etapu

**Gem** – klasa odpowiada za tworzenie na planszy diamentów generujących punkty

**8.Linki:**

1. <https://jquery.com/>
2. <http://www.html5quintus.com/>
3. http://student.agh.edu.pl/~scirka/tipgk