Krzysztof PIANTA

{CODE WITH ME}

Zostań GAME DEVELOPEREM

Helion

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiejkolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Barbara Lepionka
Projekt okładki: Maciej Grzegorek
Grafika na okładce została wykorzystana za zgodą Shutterstock.com

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *http://helion.pl* (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/codeme
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Kody źródłowe wybranych przykładów dostępne są pod adresem:
ftp://ftp.helion.pl/przyklady/codeme.zip

ISBN: 978-83-283-4998-8

Copyright © Helion 2020

Printed in Poland.

• Kup książkę
• Poleć książkę
• Oceń książkę

• Księgarnia internetowa
• Lubię to! » Nasza społeczność

Spis treści

Rozdział 1. Wstęp do gry 2D w HTML5 7
1.1. Zaczynamy 8
1.1.1. Jeden kod na wszystkie platformy 8
1.1.2. Narzędzia oraz konfiguracja (Notepad++), tworzenie kopii 8
1.2. Klocki 8
1.2.1. Świat gry (canvas, inicjalizacja) 9
1.2.2. Pętla 9
1.2.3. Rysujemy kwadrat, kółko, trójkąt, linie 11
1.2.4. Rysujemy gracza 19
1.2.5. Skalowanie obrazków 20
1.2.6. Skalowanie świata gry i obiektów (proporcjonalnie) 21
1.2.7. Rotacja 22
1.2.8. Sterowanie klawiaturą 26
1.2.9. Ulepszone sterowanie klawiaturą 29
1.2.10. Prosta kolizja z innym obiektem, np. apteczką 30
1.2.11. Poruszanie się w kierunku, w którym obrócona jest postać (klawiatura/mysz) 37
1.2.12. Kolizja kółka z kwadratem i kółka z kółkiem 42
1.2.13. Kolizja piksel po pikselu 47
1.2.14. Odległość między obiektami 65
1.2.15. Sterowanie wrogami 65
1.2.16. Proste strzelanie 70
1.2.17. Strzelanie z obrotem (wycentrowane) 77
1.2.18. Strzelanie z obrotem (ustawienia offsetu na obrazku) 79
1.2.19. Pasek zdrowia (dowolny rozmiar) 83
1.2.20. Obsługa myszy 85
1.2.21. Menu i opcje 92
1.2.22. Animacja poklatkowa 112
1.2.23. Pseudooświetlenie 116
1.2.24. Proste efekty — własny blur, blur CSS, glitch 121
1.2.25. Mikrofon 125
1.2.26. Sterowanie kontrolerem (gamepad) 126

Kup książkę Poleć książkę

1.2.27. Zaznaczanie postaci (ruch RTS) 130
1.2.28. Prosty system cząsteczek 132
1.2.29. Układ Słoneczny (orbity) 150
1.2.30. Grawitacja i jej różne zastosowania 151
1.2.31. Ratio 16×9 176
1.2.32. Loading bar i dźwięki 178
1.2.33. Okrągły pasek postępu 194
1.2.34. Lupa 196
1.2.35. Save game i load game 197
1.2.36. Licznik FPS 204
1.2.37. Ekwipunek 207
1.2.38. Kamera 209
1.2.39. Nie rysujemy tego, czego nie widzi gracz 215
1.2.40. Ściszenie dźwięków i (lub) muzyki 221
1.2.41. Centrowanie tekstu lub obrazka 223
1.2.42. Rozjaśnianie obrazu w JavaScripcie 227
1.2.43. Clip() na przykładzie textboksa à la Final Fantasy 234
1.2.44. Prerenderowanie (bufor) 238
1.3. Budujemy grę z klocków 240
1.3.1. Struktura plików naszej gry 241
1.3.2. Tworzymy grę 241
1.3.3. Tworzymy build dla Windows, Linuksa i macOS 252
1.4. Dodatki JS 253
1.4.1. Konsola dewelopera 253
1.4.2. Lista poszczególnych błędów w JS 254
1.4.3. Sztuczki w JavaScripcie 254
1.4.4. Klonowanie (referencje) 255
1.4.5. Zasięg zmiennych 259
1.4.6. This/self 261
1.4.7. jQuery kontra JavaScript 262
1.4.8. Dobra praktyka (anonimowa funkcja, window, zmienne globalne) 264
1.4.9. Komentarze blokowy i liniowy 265
1.4.10. setTimeout i dlaczego lepiej nie używać go w grach 267
1.4.11. Dlaczego requestAnimationFrame zamiast setInterval? 268
1.5. Dodatki NW.js 269
1.5.1. Edycja package.json 269
1.5.2. Steam overlay (nakładka) 270
1.5.3. Osiągnięcia Steam (greenworks) 270
1.5.4. Lepsza wydajność NW.js 275
1.5.5. Zmiana ikony aplikacji NW.js 275
1.6. Dodatki Notepad++ 278
1.6.1. Wtyczki 278
1.6.2. Polskie znaki <meta> lub kodowanie 278
1.6.3. Konfiguracja kopii zapasowej 279

Kup książkę Poleć książkę

Rozdział 2. Wstęp do gier online 281
2.1. Zaczynamy. Prosta gra tekstowa 282
2.1.1. Konfiguracja serwera XAMPP 282
2.1.2. Świat gry 283
2.1.3. Tworzenie tabeli MySQL 287
2.1.4. Rozbudowa naszego menu 288
2.1.5. Zadania cron 295
2.2. Wstęp do gry kółko i krzyżyk online (PHP/AJAX) 297
2.2.1. Zaczynamy! 297
2.3. Wstęp do kwadracików online (PHP/AJAX) 303
2.3.1. Zaczynamy! 304

Rozdział 3. Zarządzanie jakością oprogramowania 311
3.1. Optymalizacja 312
3.1.1. 300 MB kontra 60 MB 312
3.1.2. Jsperf 313
3.2. Testowanie 313
3.2.1. Jak testować 313
3.2.2. Co zrobić, gdy nie możemy znaleźć błędu? 314
3.2.3. Gamepady mogą mieć inne wartości 314
3.2.4. Najlepiej tworzyć pod low-end 314
3.3. Telemetria — śledzenie gracza 315
3.3.1. Jakie dane są wysyłane? 315
3.3.2. Prosta telemetria 315

Rozdział 4. Projektowanie gry 317
4.1. Początek 318
4.1.1. Trudne czy łatwe gry? 318
4.1.2. Gry 2D czy 3D? 318
4.1.3. Struktura plików 318
4.1.4. Gry to zabawa, nie twórz realnych gier 319
4.1.5. Programować czy korzystać z darmowych programów do tworzenia gier? 319
4.2. Zawartość gry 319
4.2.1. Kubek też ma swoją historię 320
4.2.2. Postacie 320
4.2.3. Dźwięki otoczenia 320
4.2.4. Lockit 320
4.2.5. Przykładowy scenariusz gry 321
4.2.6. Ostatni nabój — manipulowanie graczem 323
4.2.7. Licencje i uznania autorów na przykładzie Game Dev Tycoon oraz ich omówienie 323

Kup książkę Poleć książkę

Rozdział 5. Promocja gry 325
5.1. Gdzie promować grę 326
5.2. Na własną rękę 326
5.3. Wydawca 326
5.4. Agencje reklamowe 327
5.5. Keymailer 327
5.6. Platformy 327

Rozdział 6. Materiały promocyjne 329
6.1. Tworzenie materiałów 330
6.2. Press-kit 330
6.3. Trailer 330
6.4. Teaser 330
6.5. Strona WWW 330

Rozdział 7. Tworzenie zasobów 331
7.1. Programy do tworzenia grafiki 332
7.2. Programy do tworzenia muzyki i dźwięków 332
7.3. Tworzenie prostych dźwięków, modyfikacja 332
7.4. Usuwanie szumu 336
7.5. Skąd brać darmową muzykę, dźwięki oraz grafikę do wykorzystania komercyjnego? 338

Rozdział 8. Inne 341
8.1. Zakuj, naucz się i zapomnij… Zapomnij! 342
8.2. Kopia, kopia i jeszcze raz kopia! 342
8.3. Lista programów do tworzenia gier 342
8.4. Notepad++, użyteczne wtyczki Compare, ColorPicker 343
8.5. Steam — wgrywanie gry (konsola + skrypty) 345
8.6. GIF, MP4/AVI jako animowany obrazek APNG (opis Steam) 349

Zakończenie 351

Kup książkę Poleć książkę

Rozdział 4.

Projektowanie gry

Ważne jest, aby zaprojektować jak najwięcej szczegółów gry. W tym rozdziale napiszę nieco o projektowaniu gry.

Kup książkę Poleć książkę

4.1. Początek

Przed tworzeniem gry musimy podjąć kilka ważnych decyzji, aby później nie okazało się, że nasz projekt wyląduje w koszu z powodu błędnych założeń lub dlatego, że zaczęliśmy tworzyć grę na ślepo. Gdy od początku będziemy tworzyć grę dobrze, to warto utrzymać ten poziom do końca, inaczej gracz wyczuje nierówności.

W większości projektów tworzenie gry bez scenariusza i opisu postaci nie będzie dobrym pomysłem. Tworzenie gry „w locie" zawsze skończy się źle.

4.1.1. Trudne czy łatwe gry?

Nie warto tworzyć w dzisiejszych czasach trudnych gier. Musisz pamiętać, że im łatwiejsza gra, tym więcej odbiorców. Tak naprawdę nikt z nas nie gra w trudne gry.

Obecne gry mają szereg wspomagających skryptów, które wspierają gracza lub nawet czasami grają za niego. Pewne gry znanej polskiej firmy zawierają skrypty, które pokazują graczom, jak mają grać, i właściwie rozwiązują za nich zagadki… Moja gra Play with Me też to ma.

Najnowsza odsłona Tomb Raidera ustawia nawet graczowi kamerę, aby wiedział, gdzie ma podążać…

Ale gdybyś jednak się uparł, to pamiętaj, aby stworzyć dwa lub trzy poziomy trudności do wyboru przed grą. W dzisiejszych czasach gracze chętnie wybierają tryb normalny i ten tryb powinien być stworzony pod kątem casual playera, a nie hardcore playera. Z doświadczenia wiem, że i tak warto dzisiaj blokować wszystkie tryby gry oprócz normalnego. Pozostałe poziomy trudności powinny się odblokować dopiero po ukończeniu gry przez gracza.

4.1.2. Gry 2D czy 3D?

Nie ma dzisiaj większej różnicy, gry w pewnym sensie jakby od lat stały w miejscu. Graczy tak naprawdę nie interesuje grafika, lecz grywalność, albo może „oryginalność" będzie lepszym słowem.

Dobrym przykładem jest Last Doors albo Minecraft. Te gry raczej pięknej grafiki nie mają. Oczywiście gry w 3D mają przewagę, bo dają większą immersję, a dodatkowo stworzenie ładnie wyglądającej gry w 3D zawsze będzie bardziej zachwycać niż nawet najładniejsza gra w 2D, ponieważ w grach 3D grafika przypomina rzeczywistość, a przynajmniej taki jest cel ich twórców.

Gry 3D są o wiele bardziej skomplikowane, dlatego proponuję zacząć od gier 2D, a w przypadku gier 3D korzystać już z gotowych silników. Gdyby ta książka się spodobała, to napiszę taką samą o grach 3D.

4.1.3. Struktura plików

Kiedy tworzymy projekt, warto mieć wszystko poukładane, nie polecam trzymać wszystkiego w jednym folderze. Struktura plików w moich projektach wygląda następująco:

```
Nazwa_projektu/
    snd/ ← pliki dźwiękowe
    music/ ← pliki muzyczne
    js/ ← pliki javascript .js
    icons/ ← pliki z ikonami gry
```

```
libs/ ← biblioteka Greenworks (Steam)
 fonts/ ← czcionki
img/ ← pliki graficzne
    source/ ← źródła plików graficznych
    maps/ ← tworzę dodatkowo jeszcze podfoldery…
    character/ ← …trzymam osobno dekoracje mapy, postacie itd.
docs/ ← w tego typu folderach trzymam różne elementy gry…
    game/ ← jak np. pomysły, plany, opisy dotyczące gry/projektu…
    examples/ …i tworzę przykłady, które później wrzucam do gry
```

Warto zawsze mieć pliki źródłowe i trzymać je w osobnym folderze. Najczęściej chodzi o pliki graficzne, ponieważ często w nich coś zmieniamy. Jeśli zapiszemy plik już finalnie, to nie będziemy mogli edytować w nim zmian później. Pliki źródłowe w programie GIMP mają rozszerzenie *.xcf*. Pozwalają one na edytowanie pliku graficznego, możemy w nim zmieniać wszystko w dowolnej chwili.

4.1.4. Gry to zabawa, nie twórz realnych gier

Dlaczego gramy w gry? Bo sprawiają, że odrywamy się od świata rzeczywistego, od problemów i na wiele godzin możemy o nich naprawdę zapomnieć. Dlatego nigdy nie twórz gier, które są oparte na naszym realnym świecie, nasz realny świat jest nudny. Ale to nie oznacza, że nie możesz tworzyć np. symulatora mechanika; to oznacza, że taka gra ma pozostać grą. Ma mieć szereg ułatwień, to ma być zabawa.

Spójrz na GTA. Można w tej grze zrobić dużo, ale ile tych rzeczy byłbyś w stanie wykonać w rzeczywistości? I dalej żyć? No właśnie, twórz gry, które bawią graczy.

4.1.5. Programować czy korzystać z darmowych programów do tworzenia gier?

Jeśli chcesz tworzyć gry, a ta książka nie zachęciła Cię w żaden sposób do programowania, to na pewno warto skorzystać z gotowych programów pozwalających tworzyć gry. W podrozdziale 8.1 podaję listę takich programów, na pewno znajdziesz coś dla siebie. Wymienione tam programy przyśpieszą i ułatwią pracę nad grami, co nie oznacza, że zawsze będzie z górki, bo programując czy używając programów, zawsze będziesz trafiał na problemy. To, o czym warto pamiętać przed wyborem programu do tworzenia gier, to:

1. **Czy program jest popularny?** Będzie miało to wpływ na liczbę poradników, dodatków, skryptów dostępnych w sieci. Im większa społeczność, tym większe szanse na pomoc w przyszłości, gdybyśmy mieli jakieś problemy z programem.
2. **Jak wygląda licencja?** Czy program umożliwia tworzenie gier także do celów komercyjnych? Jeśli tak, to na jakich warunkach.

4.2. Zawartość gry

Gra musi posiadać tak zwany „kontent”, czyli przeróżnej maści poukrywane elementy, które nawiązują nieznacznie do innych gier, filmów lub osób, sekretne pokoje czy różne rodzaje broni i przedmiotów, w tym takie, z którymi gracz może wchodzić w interakcje, jak drzwi, okna czy inne

Kup książkę Poleć książkę

drobne elementy, które może podnieść, aby je na przykład tylko obejrzeć. Gracz nie może się nudzić i musi mieć urozmaicenie, więc warto o to zadbać.

Warto tworzyć różne rodzaje broni, ponieważ nawet jeśli od strony technicznej mamy wrażenie, że robimy po prostu to samo, bo tylko modyfikujemy kilka zmiennych, dla gracza będzie miało to znaczenie.

Wyobraź sobie najlepszą grę FPS lub RPG, w którą grałeś, i że przez cały czas używasz jednej broni. Byłoby nudno, prawda? Dlatego ważne jest, aby umieszczać jak najwięcej przedmiotów. I choć tak naprawdę następna broń różni się tylko wyglądem i innymi statystkami czy zmiennymi, to dodaje grze urozmaicenia i zmienia często także taktykę czy styl gry gracza, a to się przekłada na całość gry.

4.2.1. Kubek też ma swoją historię

Warto pamiętać, że w grach RPG czy często w grach FPS, np. horrorach, liczy się często „prawdziwość" świata. Ważne jest, czy gracz w niego uwierzy, czy nie. Postacie, żywe lub martwe, muszę mieć swoją historię, choć nie musi być ona bardzo szczegółowa.

Dla przykładu: jeśli stworzyliśmy horror i naszym miejscem akcji jest budynek jakiejś korporacji, a na podłodze leży nieżywa postać, która pracowała do późna, np. sprzątała biurowce, to warto dodać w pobliżu szczotkę czy odkurzacz, aby ją nieco uwiarygodnić. Dodatkowo można w pobliżu gdzieś zostawić jakąś notatkę o niej.

4.2.2. Postacie

Tak jak już wspomniałem, postacie powinny mieć swoją historię i przeszłość. Warto przed tworzeniem gry opisać wszystkie postacie, wiek, kolor włosów, co robią, gdzie pracują, aby uwiarygodnić je, bo dzięki temu gracz łatwiej uwierzy w historię świata gry.

Tu warto wspomnieć jeszcze o bohaterach naszej gry. Gdy na ekranie jest ich za dużo, to gracz nie jest w stanie utożsamiać się z żadną z postaci, więc duża liczba bohaterów to nie jest dobry pomysł.

4.2.3. Dźwięki otoczenia

Dzisiejsze gry mają już nieco inne standardy. Kiedyś w tle leciała (lub nawet nie) muzyka. Dzisiaj gracze są przyzwyczajeni do wysokobudżetowych produkcji, gdzie dźwięki otoczenia są standardem. Warto o to zadbać i nie ma znaczenia, czy jest to gra 2D, czy 3D (choć tutaj jest to bardziej wskazane).

Zauważyłem, że gracze coraz częściej zwracają uwagę na wydawane przez coś lub kogoś dźwięki w grach 3D i na ogół nie podoba im się, że dźwięki są po prostu generowane i odtwarzane w danej chwili, czyli jest jakiś hałas, bo np. ktoś chodzi (niby w pobliżu), ale gracz nikogo nie widzi.

4.2.4. Lockit

Co to jest lockit? To inaczej plik z tłumaczeniem gry, który w łatwy sposób będziemy przekazywać do tłumaczenia agencjom lub nawet fanom zainteresowanym tłumaczeniem.

Kup książkę Poleć książkę

Przedstawię teraz przykładowy plik lokalizacji (*lockit*), abyś wiedział, jak wygląda. Najlepiej, aby plik lokalizacji był utworzony w formie arkusza kalkulacyjnego. Można użyć darmowego oprogramowania, które potrafi zapisać go w formacie *.xls* (*Excel*), np. *OpenOffice*, lub najlepiej skorzystać z dokumentów *Google*, oprogramowania, które jest darmowe i udostępnia eksport do formatu *.xls*, oraz, co najważniejsze, taki lockit można edytować online i może to robić wielu użytkowników naraz.

Oto przykładowy plik z innymi wersjami językowymi (rysunek 4.1):

	A	B	C	D	E
1	variable	arts	polish	english	russian
2	picklock		Wytrych	Picklock	Отмычка
3	glass_small		Szkło powiększające	Magnifying glass	Лупа
4	net		Nić pajęczyny	Cobweb	Нить паутины
5	cloth		Ścierka	Cloth	Тряпка

Strings | Items | Interface | Dialogues

Gotowy

Rysunek 4.1

Jak widać na obrazku, w kolumnach są nazwy w trzech językach. Dodatkowo są obrazki, co ułatwia tłumaczom przetłumaczenie danego słowa tak, jak chcemy, aby było przetłumaczone.

Na dole są jeszcze zakładki, takie jak interfejs czy dialogi (warto oddzielić je od siebie, bo tłumaczom wygodniej się będzie tłumaczyło), które oddzielają poszczególne elementy gry do tłumaczenia.

Warto jeszcze dodać, że zawsze pierwszą kolumną jest nasza zmienna (której używamy w grze, aby wczytać odpowiedni język), a dalej dopiero tekst.

4.2.5. Przykładowy scenariusz gry

Przedstawię tutaj fragment scenariusza wymyślonej gry, który możesz wykorzystać jako szablon, a także użyć go do napisania własnego. To są szczegóły, być może oczywiste, ale gdy robisz coś pierwszy raz, to nawet najprostsze rzeczy sprawiają kłopot, warto więc mieć jakiś wzór.

Kup książkę Poleć książkę

Dodatkowo taki scenariusz, nawet średniej jakości, warto mieć, zanim zabierzemy się za tworzenie naszej gry. Ponieważ w innym przypadku będziemy dokładać pewne elementy fabuły w trakcie gry, a to w konsekwencji może spowodować mały chaos w grze, wiele rzeczy trzeba będzie zmieniać „w locie". A oto scenariusz do mojej gry Play with Me, który możesz później rozbudować:

Listing 4.1. Przykładowy scenariusz

```
Intro
Tytuł #1

Dnia 12 grudnia 1999 roku dziennikarz śledczy Robert Hawk wybrał
się z żoną, aby uczcić swoje urodziny. W czasie powrotu
do domu wydarzyło się jednak coś, co zmieniło ich życie na zawsze...

Scena #1
Scena z wypadkiem, Robert jest ciągnięty przez Iluzję po ziemi.

Tytuł #2
Robert ocknął się w ciemnym pomieszczeniu.
Jedynym znakiem, że jeszcze żyje,
był przejmujący ból głowy...

Level 1:

Napisy początkowe.

ROBERT (do siebie, lekko skołowany, zagubiony)
Ach... Moja głowa… Deszcz, chyba miałem wypadek.

ROBERT (mamrocze)
„Dziś są twoje urodziny, musimy wyskoczyć" - po co dałem się na to namówić...

ROBERT (rozpaczliwy krzyk, sam do siebie)
Sara? Sara! Jest tu kto?! Gdzie ja jestem? Co to za napis?

Robert zapala światło. Gracz sam zwraca uwagę na postać lub nie.

Robert widzi kajdanki na swoich rękach.

ROBERT (do siebie, niespokojny głos, przestraszony)
Co to za mechanizm? Nie mogę się ruszyć.

ROBERT (do siebie, niespokojny głos)
Jestem przykuty do jakiejś rury... Mogę tylko patrzeć na coś z daleka...

[Pojawia się Quest: „Uwolnij się z łańcuchów".]

Robert uwalnia się z kajdanek. Sprawdza wiszący na ścianie artykuł.

Robert sprawdza wiadomość na telefonie. To od Patricka Clarka.

Wiadomość SMS:
„Robert, gdzie jesteś?! Nie mogę się do ciebie dodzwonić... Możliwe, że w ciągu
godziny będę wiedział, kim jest Illuzja".
```

Kup książkę Poleć książkę

4.2.6. Ostatni nabój — manipulowanie graczem

Mało kto wie, że wiele gier manipuluje graczem. W wielu grach FPS, na przykład w grze Bioshock, ostatni wystrzelony nabój zabiera więcej energii naszemu przeciwnikowi, a gdy gracz ma mało HP i w ostatniej chwili zabije wroga, musi czuć się wspaniale, prawda? Tak się manipuluje graczami — im więcej emocji, tym lepiej.

Inny przykład to gra F.E.A.R. Autorzy stwierdzili, że wybuchy są fajne (więcej się dzieje na ekranie), więc wiele pocisków wystrzelonych przez gracza jest przyciąganych przez elementy, które mogą wybuchnąć; to jak magnez, który przyciąga metal.

Gry typu piłka nożna często bazują na emocjach. Fajnie jest zagrać mecz, w którym pada dużo bramek w ostatniej chwili w dramatycznych okolicznościach. Takie coś wykorzystano np. w grze FIFA (choć w ostatnich odsłonach twórcy chyba nieco przesadzili, bo gracze nie byli zadowoleni, gdy komputer jakimś cudem, przegrywając 0:2, wygrywał cały mecz 3:2).

Manipulowanie graczem to przydatna umiejętność, gracz nie zna Twojej gry. Kiedyś to zrobiłem w swojej grze Play with Me. W drugim pomieszczeniu dałem graczom do zrozumienie, że jeśli nie opuszczą tego pokoju, zamarzną. A gdy oglądałem transmisje prowadzone z gry, okazywało się, że gracze potrafili nawet wychodzić z pomieszczenia i wracać do pierwszego, aby się nieco ogrzać… Termometr w pomieszczeniu funkcjonował dobrze i pokazywał, jak temperatura spada do –30… Ale nic się nie działo. Gracze o tym nie wiedzieli i zachowywali się tak, jakby ich postać miała zaraz umrzeć z zimna.

Takie manipulowanie to dobra rzecz, ale trzeba to robić umiejętnie. To trochę jak magik na scenie i jego iluzje — gdy tylko zobaczymy od kuchni, jak daną sztuczkę się wykonuje, czar pryska.

4.2.7. Licencje i uznania autorów na przykładzie Game Dev Tycoon oraz ich omówienie

NW.JS można wykorzystać także w celach komercyjnych. NW.JS zawiera plik *credit.html*, gdzie znajduje się lista wszystkich bibliotek (wraz z informacjami o autorach), z jakich korzysta aplikacja NW.JS. Pamiętaj, aby go dołączyć do swojej aplikacji, jeśli będziesz korzystał z NW.JS, aby przestrzegać licencji. W przeciwnym razie Twoja gra może zostać zablokowana w sklepie na jakiś czas.

Chciałbym teraz opisać pokrótce niektóre licencje, na jakie możesz trafić, przeszukując internet w poszukiwaniu darmowych zasobów do gry. Pamiętaj, że zawsze możesz się skontaktować z autorem i zapytać go o szczegóły.

MIT

Jedna z najbardziej liberalnych licencji, otrzymujesz nieograniczone prawo do kopiowania, modyfikowania, rozpowszechniania (w tym sprzedaży) oraz używania oryginalnego lub zmodyfikowanego programu w postaci binarnej lub źródłowej.

Kup książkę Poleć książkę

CC 3.0 unported

Bardzo popularna licencja ze względu na jasne zasady. A ponieważ jest ich sporo, twórcy przygotowali fajną stronę, która w bardzo krótki sposób tłumaczy, co możesz, a czego nie. Autor, który korzysta z tej licencji, zawsze musi dołączyć link do odpowiedniej wersji, w tym przypadku mowa o CC 3.0 unported, czyli *https://creativecommons.org/licenses/by/3.0/*.

Licencja ta pozwala Ci na korzystanie z danego utworu (np. muzyczki, grafiki itp.), możesz modyfikować, zmieniać i wykorzystywać go w celach komercyjnych, natomiast koniecznie musisz podać link do licencji oraz autora.

Public Domain/ CC0 1.0 Universal

Autor przekazał daną rzecz domenie publicznej i zrzekł się praw autorskich, co oznacza, że nie musisz podawać autora. Dany utwór (np. grafikę, muzykę itp.) możesz wykorzystywać w celach komercyjnych, także je modyfikować.

Apache 2.0

Licencja Apache pozwala na wykorzystywanie w celach komercyjnych, modyfikowanie i redystrybucję programu w obu postaciach (źródłowej i binarnej), nie trzeba udostępniać kodu źródłowego. Twórcy oprogramowania są zwolnieni z odpowiedzialności za ich funkcjonowanie.

BSD

Ostatnimi czasy wiele osób zamiast z licencji BSD proponuje korzystanie z licencji MIT/X11 ze względu na aż trzy różne jej wersje. Można jej używać w celach komercyjnych.

GPLv3

Jest to najpopularniejsza wolna licencja. Przede wszystkim stawia na wolność, a nie na prawa użytkownika. W wielu przypadkach można jej używać w celach komercyjnych.

Wiele informacji o licencjach można znaleźć także w sieci, np. na Wikipedii, albo należy szukać artykułów na ten temat na stronie *https://www.gnu.org/licenses/license-recommendations.html*. **Pamiętaj**, aby zawsze dołączać informacje o autorze oraz licencji!

Kup książkę Poleć książkę

PROGRAM PARTNERSKI

GRUPY HELION

1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!
http://program-partnerski.helion.pl

GRUPA Helion

PROJEKTUJ, PROGRAMUJ, PROMUJ!
ZOSTAŃ TWÓRCĄ GIER KOMPUTEROWYCH!

Nie zaglądaj tu, nie warto! Stracisz tylko czas, na sto procent nie dowiesz się niczego ciekawego, znudzisz się i będziesz rozczarowany, bo... z pewnością nie chcesz dołączyć do prawdziwej elity programistów, zdobyć poszukiwanych na rynku umiejętności, nauczyć się czegoś naprawdę ekscytującego ani uzyskać wpływu na jedną z najdynamiczniej rozwijających się gałęzi przemysłu komputerowego, prawda?

Jeśli jednak mocno pragniesz zostać twórcą gier komputerowych, dobrze trafiłeś! Ta książka powstała właśnie z myślą o tych, którzy chcą rozpocząć karierę profesjonalnego game developera. Bezboleśnie wprowadzi Cię w zagadnienia związane z tworzeniem gier sieciowych 2D w językach: HTML5, PHP i MySQL. Nauczysz się projektować oprogramowanie, dbać o jakość rozwiązania, opracowywać niezbędne materiały graficzne i dźwiękowe, a nawet promować i sprzedawać swoje dzieło.

- Niszczenie terenu jak w grach Worms i Soldat
- Scrollowanie obrazu (kamera 2D)
- Pseudooświetlenie (2D lighting)
- Manipulowanie pikselami (getImageData) i proste efekty, na przykład blur (rozmycie)
- Różne typy kolizji, perfekcyjna kolizja (pixel perfect collision)
- System cząsteczek (efekty 2D): efekt gwiezdny (starfield effect), deszcz, śnieg, deszcz 3D, mgła lub dym
- NW.js (node-webkit)
- Rysowanie prostych kształtów, obrazków i sprite'ów
- Własny loader plików
- Grawitacja
- Menu obsługiwane za pomocą klawiatury lub myszy

<ZRÓB PIERWSZY KROK NA DRODZE DO PROFESJONALNEJ KARIERY!>

Helion
helion.pl
HELION SA
ul. Kościuszki 1c
44-100 Gliwice
tel.: 32 230 98 63
helion@helion.pl
INFORMATYKA W NAJLEPSZYM WYDANIU

Sprawdź nasze szkolenia!
SZKOLENIA
AKADEMIA IT & BUSINESS
HELIONSZKOLENIA.PL
KOD KORZYŚCI
Sięgnij po więcej!

ISBN 978-83-283-4998-8
9 788328 349988
Cena: 69,00 zł