

Spis treści

	O autorach	13
	Podziękowania	15
	Wprowadzenie	17
Część I	Światy dwuwymiarowe	
Rozdział 1.	Zaczynamy	23
	Narzędzia	24
	Co jest nam potrzebne?	24
	Java Development Kit	27
	Zintegrowane środowisko programistyczne	27
	Serwer stron internetowych (WWW)	33
	SDK systemu Android i emulator	34
	Techniki	38
	Kod	38
	Programowanie zorientowane obiektowo	43
	Słowo o przeglądarkach mobilnych	51
	Podsumowanie	51
Rozdział 2.	Grafika w przeglądarce — płótno	53
	Gry od podszewki	53
	Rysowanie wewnątrz przeglądarki	55
	Podstawowe ustawienia strony HTML	55
	Czym jest płótno?	56
	Kontekst	57
	Układ współrzędnych	58

Rysowanie kształtów	61
Prostokąty	62
Ścieżki	63
Podścieżki	71
Kontury i wypełnienia	72
Kolory jednolite	73
Gradienty	73
Wzorce	77
Stan kontekstu a przekształcenia	79
Przesunięcie	80
Skalowanie	81
Obrót	82
Grupowanie transformacji	82
Stan kontekstu	84
Przekształcenia kontekstu w praktyce	85
Widok przykładowego projektu — efekt końcowy	86
Podsumowanie	89
Rozdział 3. Tworzymy pierwszą grę!	91
Szkielet gry HTML5	92
Podstawowy szkielet	92
Wymuszanie orientacji	96
Architektura gry	98
Tworzymy grę	99
Renderowanie planszy	100
Stan i logika gry	105
Scalanie komponentów — klasa Game	110
Dodanie gry do szkieletu HTML	113
Podsumowanie	115
Rozdział 4. Animacje i sprite'y	117
Sprite'y	118
Wczytywanie obrazków	119
Rysowanie obrazka	129
Arkusze sprite'ów	133
Podstawy animacji	136
Najprostsza animacja	136
Wątki a JavaScript	137
Timery	138
Poprawianie jakości animacji	141
Podsumowanie	153
Rozdział 5. Obsługa zdarzeń i zachowań użytkownika	155
Zdarzenia w przeglądarce	156
Przeglądarka stacjonarna a wejście przeglądarki systemu Android	156
Stosowanie zdarzeń do przechwytywania wejścia użytkownika	157

Co jeszcze kryje się za zdarzeniami?	160
Sposób obsługi różnic pomiędzy interfejsami dotykowymi a myszą	162
Własne zdarzenia	165
Generatory i funkcje nasłuchujące własnych zdarzeń	166
EventEmitter — klasa bazowa	166
Zdarzenia a wywołania zwrotne	169
Własne zdarzenia	170
Implementacja klasy InputHandlerBase	173
Tworzymy klasę MouseInputHandler	176
Tworzymy klasę TouchInputHandler	178
Zaawansowane operacje wejściowe	179
Przeciągnij i upuść	179
Doskonale wybieranie a maski obrazków	181
Złożone operacje	182
Symulowanie dżojstika	185
Podsumowanie	188
Rozdział 6. Renderowanie wirtualnych światów	189
Mapy z kafelków	189
Zasada działania map kafelkowych	190
Implementacja mapy kafelków	191
Mierzmy FPS	196
Optymalizacja wydajności renderowania	198
Rysuj tylko to, co potrzebne	198
Bufor pozaekranowy	200
Przechowywanie okolic obszaru widoku w pamięci	203
Obiekty świata gry	206
Układy współrzędnych	206
Implementacja klasy WorldObjectRenderer	208
Porządek sortowania	211
Optymalizacje	213
Widok izometryczny	214
Podsumowanie	217
Rozdział 7. Tworzymy silnik izometryczny	219
Konfiguracja	220
Plan	221
Przygotowanie przestrzeni roboczej	222
Podstawowy kod	222
Przydatne mechanizmy	225
Teren izometryczny	231
Układy współrzędnych	231
Renderowanie kafelków	232
Implementowanie klasy IsometricTileLayer	237

Renderowanie obiektów	244
Implementacja klastrów obiektów	248
Pamięć podręczna obiektów	250
Obsługa ruchu	253
Obiekty złożone	255
Warstwa obiektów — kolejne czynności	257
Brudne prostokąty	257
Jak to działa?	258
Implementacja	259
Integracja z warstwami	263
Oznaczanie brudnych prostokątów	266
Interfejs użytkownika a menedżer warstw	268
Menedżer warstw	268
Interfejs użytkownika	271
Interakcja	274
Propagowanie i obsługa zdarzeń	275
Zatrzymywanie propagowania zdarzeń	278
Podsumowanie	279

Część II Światy trójwymiarowe

Rozdział 8. 3D w przeglądarce	281
Renderowanie 3D — wprowadzenie	282
Jak działa renderowanie 3D?	283
Przekształcenia matematyczne	283
Przykład w trójwymiarze	283
Uproszczony silnik 3D	285
Model i scena	286
Renderowanie	287
Podsumowanie	299
Rozdział 9. WebGL	301
Podstawy WebGL	302
Inicjalizacja WebGL	302
Geometrie	304
Potok renderowania OpenGL ES 2.0	305
Stosowanie buforów	307
Shadery i GLSL	308
Prosty przykład — renderujemy sześciąt	313
Zastosowanie shaderów na stronie internetowej	313
Renderowanie aplikacji typu Witaj, świecie!	315
Eksplorowanie świata WebGL	319
Kolor	319
Tekstury	324
Ładowanie modeli	327
Podsumowanie	330

Część III Łączenie światów

Rozdział 10. Pora na serwer	331
Podstawy Node.js	332
Wprowadzamy bibliotekę Node.js	332
Jak programować w Node.js?	333
Instalacja Node.js	335
Debugowanie skryptów Node.js	336
Pisanie skryptów w Node.js	337
Wyjątki a stos wywołań	338
Globalna przestrzeń nazw a moduły Node	338
Tworzymy pierwszy moduł	341
Wykrywanie modułów	344
Stosowanie NPM	345
Node.js w praktyce — tworzymy serwer dla gry	347
Frameworki do tworzenia aplikacji webowych w Node	347
Najpierw podstawy	348
Renderowanie stron internetowych	351
Parsowanie wejścia użytkownika	355
Stosowanie sesji	357
Warstwa pośrednia	358
Zachowanie porządku	360
Zgłaszanie błędów	360
Obsługa dziennika zdarzeń	364
Konfiguracje serwera	366
Podsumowanie	368
Rozdział 11. Rozmawiamy z serwerem	369
Ewolucja komunikacji sieciowej w przeglądarkach	369
Konfiguracja serwera	371
XMLHttpRequest API do obsługi żądań HTTP	372
Obsługa żądania HTTP przy użyciu XHR	373
Obsługa błędów w XHR	374
XMLHttpRequest Level 2	375
Obsługa danych binarnych	376
Odwrotny Ajax	378
Problem	378
Rozwiązania	378
Najlepsze rozwiązania	379
Dopuszczalne rozwiązania	380
Przestarzałe rozwiązania	385
Testujemy transporty	386
DDMS	386
Symulowanie złych warunków sieci przy użyciu zewnętrznego oprogramowania	387
Podsumowanie	388

Rozdział 12. Tworzymy grę sieciową	389
Architektura gry sieciowej	389
Architektura gry — od jednego do wielu graczy	391
Struktura projektu	393
Lobby w grze przy użyciu Socket.IO	394
Komunikacja klient-serwer	395
Dodawanie ekranu lobby do gry	397
Dodawanie obsługi rozgrywki	402
Współdzielenie logiki pomiędzy klientem a serwerem	403
Warstwa serwerowa	404
Warstwa kliencka	408
Podsumowanie	414
 Część IV Usprawnianie światów	
Rozdział 13. Sztuczna inteligencja w grach	415
Czy potrzebuję AI w mojej grze?	416
Wyszukiwanie ścieżek	416
Grafy	418
Czym jest graf?	418
Implementacja grafów w języku JavaScript	420
Algorytmy wyszukiwania ścieżek	424
Algorytm A*	424
Metody tworzenia grafu wyszukiwania ścieżek	430
Podejmowanie decyzji	432
Podsumowanie	435
 Rozdział 14. Silniki gier w języku JavaScript	437
API graficzne, biblioteki i silniki gier	438
API graficzne	438
Biblioteki graficzne	439
Silniki gier	440
Crafty	441
System komponentów encji	441
Witaj, świecie! w Crafty	444
Tworzymy grę w Crafty	447
Ostateczna wersja	453
Podsumowanie	455
 Rozdział 15. Tworzenie natywnych aplikacji	457
Aplikacje natywne	458
Konfiguracja Apache Cordova (PhoneGap)	459
Konfiguracja narzędzia Cordova	460
Konfiguracja Apache Ant	460

Budowanie aplikacji natywnej	461
Tworzenie pustego projektu dla systemu Android	461
Testowanie szkieletu aplikacji Android	462
Podstawowy projekt Cordova	463
Obsługa sieci	467
Ostatnie modyfikacje: nazwa, ikona i tryb pełnoekranowy	468
Stosowanie natywnych API	471
Przygotowanie do publikacji	473
Podpisywanie aplikacji	474
Publikowanie w serwisie Google Play	476
Aktualizacja aplikacji	479
Podsumowanie	481
Rozdział 16. Obsługa dźwięków	483
Dźwięki na stronach internetowych	484
Znacznik audio	484
Web Audio API	485
Dźwięk w domyślnej przeglądarce systemu Android	486
Stosowanie biblioteki SoundManager2	487
Wstępna konfiguracja	487
Odtwarzanie w pętli	489
Obsługa dźwięku w grze	490
Odtwarzanie dźwięków w aplikacjach Cordova	493
Doświadczenie użytkownika	493
Podsumowanie	494
Co dalej?	494
Dodatek A. Debugowanie aplikacji klienckich w JavaScriptcie	495
Przykład do zdebugowania	495
Debugowanie w przeglądarce stacjonarnej	496
Debugowanie urządzeń mobilnych	499
Rejestrowanie informacji (niemal) bez zmian w kodzie	500
weinre	502
Podsumowanie	503
Skorowidz	505