

React 17 : wzorce projektowe i najlepsze praktyki : projektowanie i rozijanie nowoczesnych aplikacji internetowych / Carlos Santana Roldán. – Wydanie III. – Gliwice, copyright © 2022

Spis treści

O autorze	11
O korektorach merytorycznych	13
Przedmowa	15
Część I. Witaj, React!	21
Rozdział 1. Pierwsze kroki z biblioteką React	23
Wymagania techniczne	24
Różnica między programowaniem imperatywnym i deklaratywnym	25
Sposób działania elementów biblioteki React	27
Zapomnieć o wszystkim	28
Problem znużenia kodem w JavaScriptcie	30
Wprowadzenie do języka TypeScript	33
Elementy języka TypeScript	33
Przekształcanie kodu w JavaScriptcie w kod w TypeScriptcie	34
Typy	35
Interfejsy	36
Podsumowanie	39
Rozdział 2. Czyszczenie kodu	41
Wymagania techniczne	41
Zastosowanie składni JSX	42
Babel 7	43
Tworzenie pierwszego elementu	44
Elementy modelu DOM i komponenty biblioteki React	44
Własności	45
Elementy potomne	45
Różnice względem języka HTML	46
Atrybuty rozwinięcia	50
Literały szablonów	50
Typowe wzorce	50
Określanie stylu kodu	58
EditorConfig	59
Prettier	59
ESLint	60
Programowanie funkcyjne	64

Funkcje pierwszoklasowe	65
Czystość	65
Niezmiennosc	66
Rozwijanie funkcji	67
Kompozycja	67
Programowanie funkcyjne i interfejsy użytkownika	67
Podsumowanie	68
Część II. Działanie biblioteki React	69
Rozdział 3. React Hooks	71
Wymagania techniczne	71
Wprowadzanie do dodatku React Hooks	72
Bez przełomowych zmian	72
Zastosowanie funkcji Hook useState	72
Reguły funkcji Hook	73
Migracja komponentu klasowego pod kątem użycia dodatku	
React Hooks	74
Efekty biblioteki React	78
Funkcja Hook useEffect	78
Warunkowe aktywowanie efektu	78
Funkcje useCallback, useMemo i memo	79
Zapamiętywanie komponentu za pomocą funkcji memo	82
Zapamiętywanie wartości za pomocą funkcji useMemo	84
Zapamiętywanie definicji funkcji za pomocą funkcji useCallback	87
Zapamiętywanie funkcji przekazanej jako argument funkcji useEffect	92
Funkcja Hook useReducer	95
Podsumowanie	99
Rozdział 4. Przegląd popularnych wzorców kompozycji	100
Wymagania techniczne	100
Zapewnienie komunikacji między komponentami	101
Zastosowanie własności children	101
Wzorce kontenerowe i prezentacyjne	103
Komponenty wyższego rzędu	108
Komponent FunctionAsChild	111
Podsumowanie	112
Rozdział 5. Omówienie języka GraphQL na przykładzie rzeczywistego projektu	113
Wymagania techniczne	114
Instalowanie systemu baz danych PostgreSQL	114
Najlepsze narzędzia do zarządzania bazą danych PostgreSQL	115
Tworzenie pliku .env i plików konfiguracyjnych	116
Konfigurowanie pliku .env	116
Tworzenie podstawowego pliku konfiguracyjnego	117

Konfigurowanie serwera Apollo Server	118
Definiowanie typów, zapytań i przekształceń języka GraphQL	120
Zapytania	121
Przekształcenia	121
Scalanie definicji typów	122
Tworzenie resolverów	122
Tworzenie zapytania getUsers	123
Tworzenie zapytania getUserData	124
Tworzenie przekształceń	125
Scalanie resolverów	125
Tworzenie modeli narzędzia Sequelize	126
Nawiązywanie połączenia przez narzędzie Sequelize z bazą danych PostgreSQL	127
Funkcje uwierzytelniania	128
Czym jest standard JSON Web Token?	128
Funkcje JWT	128
Tworzenie funkcji uwierzytelniania	130
Typy i interfejsy	131
Uruchamianie projektu po raz pierwszy	133
Testowanie zapytań i przekształceń języka GraphQL	135
Sprawdzanie poprawności	139
Wykonywanie operacji logowania	142
Tworzenie interfejsowego systemu logowania z wykorzystaniem klienta Apollo	146
Konfigurowanie narzędzia Webpack 5	146
Konfigurowanie interpretera języka TypeScript	149
Konfigurowanie serwera Express	150
Tworzenie konfiguracji interfejsu	151
Tworzenie elementu pośredniczącego użytkownika	153
Tworzenie funkcji tokenów JWT	155
Tworzenie zapytań i przekształceń języka GraphQL	156
Tworzenie kontekstu użytkownika do obsługi operacji logowania i połączonego użytkownika	157
Konfigurowanie klienta Apollo Client	159
Tworzenie ścieżek aplikacji	159
Tworzenie stron	160
Tworzenie komponentów procesu logowania	161
Tworzenie komponentów panelu sterowania	164
Testowanie systemu logowania	165
Podsumowanie	169
Rozdział 6. Zarządzanie danymi	170
Wymagania techniczne	170
Wprowadzenie do składnika React Context API	171
Tworzenie pierwszego kontekstu	171
Opakowywanie komponentów za pomocą dostawcy	173

Korzystanie z kontekstu za pomocą funkcji useContext	173
Wprowadzenie do stosowania składnika React Suspense z wykorzystaniem biblioteki SWR	175
Wprowadzenie do biblioteki SWR	175
Budowanie aplikacji Pokedex!	175
Testowanie składnika React Suspense	181
Podsumowanie	185
Rozdział 7. Tworzenie kodu dla przeglądarki	186
Wymagania techniczne	187
Zastosowanie formularzy	187
Komponenty niekontrolowane	187
Komponenty kontrolowane	191
Obsługa zdarzeń	193
Referencje	196
Implementowanie animacji	197
React Motion	199
Format SVG	200
Podsumowanie	202
Część III. Wydajność, ulepszenia i środowisko produkcyjne	203
Rozdział 8. Zapewnianie ładnego wyglądu komponentów	205
Wymagania techniczne	206
Style CSS w kodzie w JavaScriptcie	206
Zastosowanie stylów lokalnych	208
Biblioteka Radium	211
Zastosowanie modułów stylów CSS	215
Webpack 5	215
Przygotowywanie projektu	216
Style CSS o zasięgu lokalnym	221
Moduły Atomie CSS	226
Moduły CSS biblioteki React	228
Zastosowanie biblioteki styled-components	229
Podsumowanie	231
Rozdział 9. Renderowanie serwerowe dla zabawy i zarobku	233
Wymagania techniczne	234
Uniwersalne aplikacje	234
Powody implementowania opcji SSR	235
Implementowanie optymalizacji pod kątem wyszukiwarki internetowej	235
Wspólna baza kodu	236
Lepsza wydajność	237
Nie lekceważ złożoności	237
Tworzenie prostego przykładu użycia opcji SSR	238
Implementowanie pobierania danych	244

Zastosowanie środowiska Next.js do tworzenia aplikacji opartych na bibliotece React	247
Podsumowanie	251
Rozdział 10. Zwiększanie wydajności aplikacji	252
Wymagania techniczne	253
Uzgadnianie	253
Klucze	254
Techniki optymalizacji	255
Narzędzia i biblioteki	257
Niezmiennosc	257
Dodatki narzędzia Babel	258
Podsumowanie	259
Rozdział 11. Testowanie i debugowanie	260
Wymagania techniczne	260
Korzyści wynikające z testowania	261
Bezproblemowe testowanie kodu w JavaScriptcie za pomocą środowiska Jest	262
Testowanie zdarzeń	267
Zastosowanie rozszerzenia React DevTools	269
Stosowanie rozszerzenia Redux DevTools	269
Podsumowanie	270
Rozdział 12. React Router	271
Wymagania techniczne	271
Instalowanie i konfigurowanie biblioteki React Router	272
Tworzenie sekcji	272
Dodawanie parametrów do ścieżek	277
Podsumowanie	284
Rozdział 13. Antywzorce, jakich należy unikać	285
Wymagania techniczne	285
Inicjalizowanie stanu za pomocą właściwości	286
Stosowanie indeksów jako klucza	288
Rozmieszczanie właściwości w elementach modelu DOM	290
Podsumowanie	292
Rozdział 14. Wdrażanie aplikacji w środowisku produkcyjnym	293
Wymagania techniczne	293
Tworzenie pierwszego Dropletu usługi DigitalOcean	294
Rejestrowanie w usłudze DigitalOcean	294
Tworzenie pierwszego Dropletu	297
Instalowanie środowiska Node.js	299
Konfigurowanie serwisów Git GitHub	300
Wyłączanie Dropletu	303

Konfigurowanie serwera nginx, narzędzia PM2 i domeny	304
Instalowanie i konfigurowanie serwera nginx	305
Konfigurowanie serwera odwrotnego proxy	306
Dodawanie domeny do Dropletu	307
Implementowanie narzędzia CircleCI do zapewnienia ciągłej integracji	309
Dodawanie klucza SSH do narzędzia CircleCI	310
Konfigurowanie narzędzia CircleCI	312
Tworzenie zmiennych środowiskowych w narzędziu CircleCI	314
Podsumowanie	318
Rozdział 15. Kolejne kroki	319
Wymagania techniczne	319
Uczestniczenie w rozwoju biblioteki React	320
Dystrybucja kodu	321
Znajomość najlepszych praktyk związanych z publikowaniem kodu open source	322
Publikowanie pakietu za pomocą narzędzia npm	324
Podsumowanie	325

oprac. BPK