
Spis treści

Wstęp	13
Wprowadzenie	15
1. Środowisko programistyczne	19
Edytor tekstu	20
Terminal	20
Użycie dedykowanej aplikacji terminala	20
Użycie Visual Studio Code	20
Poruszanie się po systemie plików	20
Narzędzia powłoki i menedżer Homebrew (tylko w systemie macOS)	21
Node.js i menedżer pakietów npm	22
Instalacja Node.js i npm w systemie macOS	22
Instalacja Node.js i npm w systemie Windows	23
MongoDB	23
Instalacja i uruchomienie MongoDB w systemie macOS	23
Instalacja i uruchomienie MongoDB w systemie Windows	24
Git	24
Expo	25
Prettier	25
ESLint	26
Ładny wygląd kodu w edytorze	26
Podsumowanie	26
2. Wprowadzenie do API	27
Co będziemy budować?	27
Jak zbudujemy nasze API?	28
Rozpoczęcie pracy	28
Podsumowanie	30

3. Aplikacja internetowa utworzona za pomocą Node.js i frameworka Express	31
Witaj, świecie!	31
nodemon	32
Rozszerzone opcje portu	33
Podsumowanie	34
4. Pierwsze API GraphQL	35
Zmiana serwera na API (mniej więcej)	35
Podstawy GraphQL	39
Schemat	39
Funkcja resolvera	40
Dostosowanie API do naszych potrzeb	41
Podsumowanie	47
5. Baza danych	49
Rozpoczęcie pracy z MongoDB	50
Połączenie MongoDB z aplikacją	51
Odczytywanie i zapisywanie danych przez aplikację	55
Podsumowanie	61
6. Operacje CRUD	63
Rozdzielenie schematu GraphQL i funkcji resolverów	63
Tworzenie schematu CRUD GraphQL	66
Funkcje resolverów CRUD	67
Data i godzina	69
Podsumowanie	71
7. Konta użytkownika i uwierzytelnianie	73
Uwierzytelnianie w aplikacji	73
Szyfrowanie i tokeny	74
Szyfrowanie hasła	74
Tokeny JSON Web	75
Integracja uwierzytelniania z naszym API	77
Schematy użytkownika	77
Funkcje resolverów uwierzytelniania	78
Dodawanie użytkownika do kontekstu funkcji resolvera	82
Podsumowanie	84
8. Akcje użytkownika	85
Zanim zaczniesz	85
Dodawanie użytkownika do nowej notatki	85
Uprawnienia użytkownika w zakresie uaktualniania i usuwania notatek	88

Zapytania użytkownika	89
Oznaczanie notatki jako ulubionej	92
Zapytania zagnieżdżone	96
Podsumowanie	98
9. Istotne szczegóły	99
Najlepsze praktyki podczas tworzenia aplikacji internetowej i aplikacji opartej na Express.js	99
Express Helmet	99
CORS	100
Stronicowanie	100
Ograniczenia danych	103
Pozostałe aspekty tworzenia naszej aplikacji	104
Testowanie	104
Subskrypcje	104
Platforma Apollo GraphQL	104
Podsumowanie	104
10. Wdrożenie naszego API	105
Hosting bazy danych	105
Wdrożenie aplikacji	109
Konfiguracja projektu	111
Wdrożenie kodu aplikacji	112
Testowanie	113
Podsumowanie	113
11. Interfejsy użytkownika i React	115
JavaScript i interfejs użytkownika	116
Interfejsy deklaratywne tworzone za pomocą JavaScriptu	117
Minimalna znajomość biblioteki React	117
Podsumowanie	122
12. Budowa klienta internetowego za pomocą React	123
Co będziemy budować?	123
Jak będziemy budować naszą aplikację?	124
Rozpoczęcie pracy	125
Budowa aplikacji	126
Routing	127
Łączenie	131
Komponenty UI	131
Podsumowanie	134

13. Nadawanie stylu aplikacji	135
Tworzenie komponentu układu	135
CSS	137
Biblioteka typu CSS-in-JS	138
Utworzenie komponentu przycisku	139
Dodawanie stylów globalnych	140
Style komponentu	142
Podsumowanie	145
14. Praca z klientem Apollo	147
Konfiguracja klienta Apollo	148
Wykonywanie zapytań do API	149
Wybrane style	156
Zapytania dynamiczne	158
Stronicowanie	161
Podsumowanie	163
15. Uwierzytelnianie i informacje o stanie	165
Utworzenie formularza rejestracyjnego	165
Formularze React i informacje o stanie	168
Mutacja signUp	170
Tokeny JWT i lokalne magazyny danych	173
Przekierowania	174
Dołączanie nagłówków do żądania	174
Zarządzanie lokalnymi informacjami o stanie	175
Wylogowanie	177
Utworzenie formularza logowania	180
Trasy chronione	185
Podsumowanie	186
16. Operacje tworzenia, odczytywania, uaktualniania i usuwania	187
Tworzenie nowych notatek	187
Odczytywanie notatek użytkownika	193
Uaktualnianie notatek	196
Usunięcie notatki	202
Zmiana ulubionych	204
Podsumowanie	208

17. Wdrożenie aplikacji internetowej	209
Statyczna witryna internetowa	209
Procedura wdrożenia	210
Hosting kodu źródłowego w repozytorium Git	211
Wdrożenie z użyciem Netlify	212
Podsumowanie	214
18. Aplikacje utworzone za pomocą frameworka Electron	215
Co zbudujemy?	215
Jak utworzymy aplikację?	215
Rozpoczęcie pracy	216
Nasza pierwsza aplikacja Electron	217
Szczegóły związane z oknem aplikacji w systemie macOS	218
Narzędzia programistyczne	219
API frameworka Electron	220
Podsumowanie	221
19. Integracja istniejącej aplikacji internetowej z frameworkiem Electron	223
Integracja aplikacji internetowej	223
Ostrzeżenia i błędy	224
Konfiguracja	226
Polityka CSP	227
Podsumowanie	229
20. Wdrożenie aplikacji frameworka Electron	231
Electron Builder	231
Konfigurowanie Electron Builder	232
Kompilacja dla bieżącej platformy	233
Ikony aplikacji	233
Kompilacja aplikacji dla różnych platform	234
Podpisywanie kodu	234
Podsumowanie	235
21. Tworzenie aplikacji mobilnych za pomocą React Native	237
Co będziemy tworzyć?	237
W jaki sposób utworzymy aplikację?	238
Rozpoczęcie pracy	239
Podsumowanie	242
22. Tworzenie aplikacji mobilnej	243
Elementy konstrukcyjne React Native	243
Style i biblioteka Styled Components	245
Biblioteka Styled Components.....	247

Routing	248
Nawigacja oparta na kartach i React Native	249
Nawigacja oparta na stosie i React Native	251
Dodawanie tytułów ekranów	255
Ikony	256
Podsumowanie	258
23. GraphQL i React Native	259
Utworzenie widoków listy i przewijanej treści	259
Zapewnienie routingu listy	265
GraphQL i klient Apollo	266
Tworzenie zapytań GraphQL	267
Dodanie paska postępu wczytywania danych	273
Podsumowanie	273
24. Uwierzytelnianie w aplikacji mobilnej	275
Mechanizm uwierzytelniania	275
Utworzenie formularza logowania	283
Uwierzytelnienie za pomocą mutacji GraphQL	288
Uwierzytelnione zapytania GraphQL	290
Dodanie formularza rejestracji	293
Podsumowanie	298
25. Dystrybucja aplikacji mobilnej	301
Konfiguracja pliku app.json	301
Ikony i ekrany wczytywania aplikacji	303
Ikony aplikacji	303
Winietka	304
Publikowanie aplikacji za pomocą Expo	305
Tworzenie kompilacji natywnych	306
iOS	306
Android	307
Dystrybucja aplikacji w sklepach Apple'a i Google'a	308
Podsumowanie	308
Zakończenie	309
A Lokalne uruchomienie API	311
B Lokalne uruchomienie aplikacji internetowej	313