

Spis treści

O autorze	12
O recenzentach	13
Przedmowa	15
Rozdział 1. Utworzenie aplikacji bloga	21
Instalacja Django	22
Utworzenie odizolowanego środowiska Pythona	23
Instalowanie Django za pomocą pip	23
Tworzenie pierwszego projektu	24
Uruchamianie serwera programistycznego	25
Ustawienia projektu	27
Projekty i aplikacje	28
Utworzenie aplikacji	28
Projekt schematu danych dla bloga	30
Aktywacja aplikacji	32
Tworzenie i stosowanie migracji	32
Tworzenie witryny administracyjnej dla modeli	34
Tworzenie superużytkownika	34
Witryna administracyjna Django	34
Dodawanie modeli do witryny administracyjnej	35
Personalizacja sposobu wyświetlania modeli	37
Praca z obiektami QuerySet i menedżerami	39
Tworzenie obiektów	39
Aktualizowanie obiektów	40
Pobieranie obiektów	41
Usunięcie obiektu	42
Kiedy następuje określenie zawartości kolekcji QuerySet?	42
Utworzenie menedżerów modelu	42



Przygotowanie widoków listy i szczegółów	44
Utworzenie widoków listy i szczegółów	44
Dodanie wzorców adresów URL do widoków	45
Kanoniczne adresy URL dla modeli	47
Utworzenie szablonów dla widoków	47
Dodanie stronicowania	50
Użycie widoków opartych na klasach	52
Podsumowanie	54
Rozdział 2. Usprawnienie bloga za pomocą funkcji zaawansowanych	55
Współdzielenie postów przy użyciu wiadomości e-mail	56
Tworzenie formularzy w Django	56
Obsługa formularzy w widokach	57
Wysyłanie wiadomości e-mail w Django	59
Generowanie formularza w szablonie	61
Utworzenie systemu komentarzy	65
Budowanie modelu	65
Utworzenie formularza na podstawie modelu	67
Obsługa klasy ModelForm w widoku	67
Dodanie komentarzy do szablonu szczegółów posta	69
Dodanie funkcjonalności tagów	72
Pobieranie podobnych postów	77
Podsumowanie	80
Rozdział 3. Rozbudowa aplikacji bloga	81
Utworzenie własnych filtrów i znaczników szablonu	82
Utworzenie własnych znaczników szablonu	82
Utworzenie własnych filtrów szablonu	87
Dodanie mapy witryny	89
Utworzenie kanału wiadomości dla postów bloga	92
Dodanie do bloga wyszukiwania pełnotekstowego	95
Instalacja PostgreSQL	95
Proste wyszukiwania	96
Wyszukiwanie w wielu polach	97
Utworzenie widoku wyszukiwania	97
Stemming i ranking wyników	100
Wagi zapytań	101
Wyszukiwanie z podobieństwem trygramu	101
Inne silniki wyszukiwania pełnotekstowego	102
Podsumowanie	102
Rozdział 4. Utworzenie witryny społecznościowej	103
Utworzenie projektu witryny społecznościowej	104
Rozpoczęcie pracy nad aplikacją społecznościową	104
Użycie frameworka uwierzytelniania w Django	105
Utworzenie widoku logowania	106
Użycie widoków uwierzytelniania w Django	111
Widoki logowania i wylogowania	112

Widoki zmiany hasła	117
Widoki zerowania hasła	119
Rejestracja użytkownika i profile użytkownika	124
Rejestracja użytkownika	124
Rozbudowa modelu User	127
Użycie frameworka komunikatów	133
Implementacja własnego mechanizmu uwierzytelniania	135
Dodanie do witryny uwierzytelnienia za pomocą innej witryny społecznościowej	137
Uruchomienie serwera programistycznego za pośrednictwem HTTPS	139
Uwierzytelnienie za pomocą serwisu Facebook	140
Uwierzytelnienie za pomocą serwisu Twitter	145
Uwierzytelnienie za pomocą serwisu Google	148
Podsumowanie	153
 Rozdział 5. Udostępnianie treści w witrynie internetowej	 155
Utworzenie witryny internetowej do kolekcjonowania obrazów	156
Utworzenie modelu Image	156
Zdefiniowanie relacji typu „wiele do wielu”	158
Rejestracja modelu Image w witrynie administracyjnej	158
Umieszczanie treści pochodzącej z innych witryn internetowych	159
Usunięcie zawartości pól formularza	160
Nadpisanie metody save() egzemplarza ModelForm	160
Utworzenie bookmarkletu za pomocą jQuery	165
Utworzenie szczegółowego widoku obrazu	172
Utworzenie miniatury za pomocą easy-thumbnails	174
Dodanie akcji AJAX za pomocą jQuery	175
Wczytanie jQuery	176
CSRF w żądaniach AJAX	177
Wykonywanie żądań AJAX za pomocą jQuery	178
Utworzenie własnych dekoratorów dla widoków	182
Dodanie stronicowania AJAX do widoków listy	183
Podsumowanie	187
 Rozdział 6. Śledzenie działań użytkownika	 189
Utworzenie systemu obserwacji	189
Utworzenie relacji typu „wiele do wielu” za pomocą modelu pośredniego	190
Utworzenie widoków listy i szczegółowego dla profilu użytkownika	193
Utworzenie widoku AJAX pozwalającego na obserwację użytkowników	197
Budowa aplikacji z ogólnym strumieniem aktywności	199
Użycie frameworka contenttypes	200
Dodanie do modelu relacji generycznych	201
Uniknięcie powielonych akcji w strumieniu aktywności	203
Dodanie akcji użytkownika do strumienia aktywności	204
Wyświetlanie strumienia aktywności	205
Optymalizacja kolekcji QuerySet dotyczącej powiązanych obiektów	206
Tworzenie szablonów dla akcji	207
Użycie sygnałów dla denormalizowanych zliczeń	209
Praca z sygnałami	209
Definiowanie klas konfiguracyjnych aplikacji	211

Użycie bazy danych Redis do przechowywania różnych elementów widoków	213
Instalacja bazy danych Redis	214
Użycie bazy danych Redis z Pythonem	215
Przechowywanie różnych elementów widoków w bazie danych Redis	216
Przechowywanie rankingu w bazie danych Redis	217
Kolejne kroki z bazą danych Redis	220
Podsumowanie	220
Rozdział 7. Utworzenie sklepu internetowego	221
Utworzenie projektu sklepu internetowego	222
Utworzenie modeli katalogu produktów	223
Rejestracja modeli katalogu w witrynie administracyjnej	225
Utworzenie widoków katalogu	226
Utworzenie szablonów katalogu	228
Utworzenie koszyka na zakupy	232
Użycie sesji Django	232
Ustawienia sesji	233
Wygaśnięcie sesji	234
Przechowywanie koszyka na zakupy w sesji	234
Utworzenie widoków koszyka na zakupy	238
Utworzenie procesora kontekstu dla bieżącego koszyka na zakupy	245
Rejestracja zamówień klienta	247
Utworzenie modeli zamówienia	247
Dołączenie modeli zamówienia w witrynie administracyjnej	249
Utworzenie zamówień klienta	250
Wykonywanie zadań asynchronicznych za pomocą Celery	254
Instalacja Celery	254
Instalacja RabbitMQ	254
Dodanie Celery do projektu	255
Dodawanie do aplikacji zadań asynchronicznych	256
Monitorowanie Celery	258
Podsumowanie	258
Rozdział 8. Zarządzanie płatnościami i zamówieniami	259
Integracja bramki płatności	259
Tworzenie konta sandbox serwisu Braintree	260
Instalowanie modułu Pythona Braintree	261
Integracja bramki płatności	262
Testowanie płatności	269
Wdrożenie do produkcji	271
Eksport zamówień do plików CSV	271
Dodanie własnych akcji do witryny administracyjnej	272
Rozbudowa witryny administracyjnej za pomocą własnych widoków	274
Dynamiczne generowanie rachunków w formacie PDF	278
Instalacja WeasyPrint	278
Utworzenie szablonu PDF	279
Generowanie pliku w formacie PDF	280
Wysyłanie dokumentów PDF za pomocą wiadomości e-mail	282
Podsumowanie	285

Rozdział 9. Rozbudowa sklepu internetowego	287
Utworzenie systemu kuponów	287
Utworzenie modeli kuponu	288
Zastosowanie kuponu w koszyku na zakupy	290
Zastosowanie kuponu w zamówieniu	295
Internacjonalizacja i lokalizacja projektu	297
Internacjonalizacja za pomocą Django	298
Przygotowanie projektu do internacjonalizacji	300
Tłumaczenie kodu Pythona	301
Tłumaczenie szablonów	306
Użycie interfejsu do tłumaczeń o nazwie Rosetta	310
Opcja fuzzy	312
Wzorce adresów URL dla internacjonalizacji	313
Umożliwienie użytkownikowi zmiany języka	315
Tłumaczenie modeli za pomocą django-parler	316
Format lokalizacji	324
Użycie modułu django-localflavor do weryfikacji pól formularza	324
Utworzenie silnika rekomendacji produktu	326
Rekomendacja produktu na podstawie wcześniejszych transakcji	326
Podsumowanie	333
Rozdział 10. Budowa platformy e-learningu	335
Utworzenie platformy e-learningu	335
Utworzenie modeli kursu	336
Rejestracja modeli w witrynie administracyjnej	338
Użycie fikstur w celu dostarczenia początkowych danych dla modeli	339
Utworzenie modeli dla zróżnicowanej treści	341
Wykorzystanie dziedziczenia modelu	342
Utworzenie modeli treści	344
Utworzenie własnych kolumn modelu	346
Dodawanie porządkowania do modułów i obiektów treści	348
Utworzenie systemu zarządzania treścią	352
Dodanie systemu uwierzytelniania	352
Utworzenie szablonów uwierzytelniania	352
Utworzenie widoków opartych na klasach	355
Użycie domieszek w widokach opartych na klasach	355
Praca z grupami i uprawnieniami	358
Zarządzanie modułami kursu i treścią	364
Użycie zbiorów formularzy dla modułów kursów	364
Dodanie treści do modułów kursów	368
Zarządzanie modułami i treścią	373
Zmiana kolejności modułów i treści	377
Podsumowanie	380
Rozdział 11. Renderowanie i buforowanie treści	381
Wyświetlanie kursów	382
Dodanie rejestracji uczestnika	386
Utworzenie widoku rejestracji uczestnika	386
Zapisanie się na kurs	389

Uzyskanie dostępu do treści kursu	392
Generowanie różnych rodzajów treści	395
Użycie frameworka buforowania	398
Dostępne mechanizmy buforowania	398
Instalacja Memcached	399
Ustawienia bufora	400
Dodanie Memcached do projektu	400
Poziomy buforowania	401
Użycie niskopoziomowego API buforowania	402
Buforowanie fragmentów szablonu	405
Buforowanie widoków	406
Podsumowanie	407
 Rozdział 12. Utworzenie API	 409
Utworzenie API typu RESTful	410
Instalacja Django Rest Framework	410
Definiowanie serializacji	411
Klasy parserów i renderowania formatów	412
Utworzenie widoków listy i szczegółowego	413
Serializacja zagnieżdżona	415
Utworzenie własnych widoków	417
Obsługa uwierzytelniania	418
Określenie uprawnień do widoków	419
Utworzenie kolekcji widoku i routerów	420
Dołączenie dodatkowych akcji do kolekcji widoku	422
Tworzenie własnych uprawnień	423
Serializacja treści kursu	423
Wykorzystanie API RESTful	425
Podsumowanie	428
 Rozdział 13. Budowanie serwera czatu	 429
Utworzenie aplikacji czatu	429
Implementacja widoku pokoju czatu	430
Dezaktywacja buforowania na poziomie witryny	432
Obsługa czasu rzeczywistego w Django za pomocą frameworka Channels	433
Aplikacje asynchroniczne z wykorzystaniem ASGI	433
Cykl żądanie-odpowiedź z wykorzystaniem frameworka Channels	434
Instalacja frameworka Channels	436
Pisanie konsumenta	438
Routing	439
Implementacja klienta WebSocket	440
Warstwa kanału komunikacyjnego	445
Kanały komunikacyjne i grupy	446
Konfiguracja warstwy kanału komunikacyjnego z wykorzystaniem Redis	446
Aktualizacja konsumenta w celu rozgłaszania wiadomości	447
Dodawanie kontekstu do wiadomości	451
Modyfikacja konsumenta w celu uzyskania pełnej asynchroniczności	454
Integracja aplikacji czatu z istniejącymi widokami	456
Podsumowanie	457

Rozdział 14. Wdrożenie	459
Zarządzanie ustawieniami dla wielu środowisk	460
Instalacja PostgreSQL	462
Sprawdzenie projektu	463
Udostępnianie Django za pomocą WSGI	463
Instalacja uWSGI	463
Konfiguracja uWSGI	464
Instalacja NGINX	467
Środowisko produkcyjne	468
Konfiguracja Nginx	468
Udostępnianie zasobów statycznych i multimedialnych	470
Zabezpieczanie połączeń za pomocą SSL/TLS	473
Wykorzystanie serwera Daphne z frameworkiem Django Channels	477
Wykorzystanie bezpiecznych połączeń dla gniazd WebSocket	478
Uwzględnienie Daphne w konfiguracji NGINX	478
Utworzenie własnego oprogramowania pośredniczącego	481
Implementacja własnych poleceń administracyjnych	484
Podsumowanie	487