Przedmowa (11)
Rozdział 1. Poznaj Laravel (15)

- Potrzeba stosowania frameworków (16)
  - Ograniczenia tworzonych przez siebie narzędzi (16)
  - Laravel przybywa na ratunek (16)
- Nowe podejście do tworzenia aplikacji w języku PHP (17)
  - Znacznie przyjaźniejsza obsługa HTTP (17)
  - Wykorzystanie pełni możliwości języka PHP (17)
- Główne funkcje Laravel i źródła ich inspiracji (18)
  - Prostota i zwięzłość (20)
  - Odpowiedzialność, nazewnictwo i konwencje (21)
  - Pomoc w staniu się lepszym programistą (22)
- Struktura aplikacji Laravel (23)
  - Kontener aplikacji i cykl życia żądania (24)
  - Poznawanie Laravel (24)
  - Migracja z wersji 3. do wersji 4. (25)
- Podsumowanie (26)

Rozdział 2. Narzędzie Composer (27)

- Korzystanie z wiersza poleceń (28)
- Jak działa Composer? (28)
- Instalacja (29)
  - Unix (Mac OS X, Linux) (29)
  - Windows (30)
- Tworzenie nowej aplikacji Laravel (30)
- Odnajdywanie i instalacja nowych pakietów (31)
- Kilka dodatkowych rad (32)
- Podsumowanie (33)

Rozdział 3. Pierwsza aplikacja (35)

- Tworzenie szkicu aplikacji (36)
  - Encje, związki i atrybuty (36)
  - Mapa aplikacji, czyli adresy URL (36)
- Uruchomienie aplikacji (37)
  - Użycie wbudowanego serwera deweloperskiego (38)
- Utworzenie pierwszych ścieżek (38)
  - Ograniczenie parametrów routingu (39)
  - Wyłapywanie brakujących ścieżek (39)
  - Obsługa przekierowań (40)
  - Zwracanie widoków (41)
- Przygotowanie bazy danych (41)
  - Tworzenie modeli Eloquent (41)
  - Budowanie schematu bazy danych (42)
  - Wstawienie danych początkowych (43)
- Szablony Blade (44)
  - Wykonanie widoku głównego (45)
- Powrót do routingu i adresów URL (46)

    - Strona podsumowania (46)
    - Wyświetlenie strony konkretnego kota (48)
    - Dodanie, edycja i usunięcie danych kota (49)
- Podsumowanie (52)

Rozdział 4. Uwierzytelnianie i bezpieczeństwo (53)

- Uwierzytelnianie użytkowników (53)
    - Tworzenie modelu użytkownika (54)
    - Utworzenie niezbędnego schematu bazy danych (54)
    - Widoki i ścieżki routingu związane z uwierzytelnieniem (56)
    - Sprawdzanie danych wejściowych (59)
- Zabezpieczanie aplikacji (60)
    - Atak typu CSRF (61)
    - Atak typu XSS (61)
    - Unikanie wstrzyknięcia kodu SQL (62)
    - Ostrożne korzystanie z masowego przypisywania wartości (63)
    - Pliki cookies - domyślnie bezpieczne (63)
    - Wymuszenie protokołu HTTPS przy wymianie danych wrażliwych (63)
- Podsumowanie (64)

Rozdział 5. Testy - to łatwiejsze, niż się wydaje (65)

- Zalety tworzenia testów (66)
- Anatomia testu (66)
- Testy jednostkowe PHPUnit (67)
    - Definiowanie oczekiwanego wyniku za pomocą asercji (68)
    - Przygotowanie sceny i wyczyszczenie obiektów (68)
    - Przygotowanie się na wyjątki (69)
    - Testowanie powiązanych ze sobą klas w pełnej izolacji (69)
- Testy integracyjne (70)
    - Testowanie - pobieranie bibliotek (70)
    - Asercje dotyczące frameworka (71)
    - Podszywanie się pod użytkowników (72)
    - Testy z użyciem bazy danych (72)
    - Sprawdzanie kodu HTML zwracanego przez widok (73)
- Podsumowanie (74)

Rozdział 6. Artisan - narzędzie wiersza poleceń (75)

- Pobieranie najnowszych zmian (75)
- Interakcja i sprawdzanie aplikacji (76)
    - Zabawa z wewnętrznymi elementami systemu (77)
    - Tymczasowe wyłączenie aplikacji (77)
    - Optymalizacja aplikacji (78)
- Instalacja poleceń innych twórców (78)
    - Przyspieszenie prac programistycznych za pomocą generatorów (78)
    - Wdrażanie aplikacji jednym poleceniem (80)
- Tworzenie własnych poleceń artisan (81)
    - Tworzenie polecenia (82)

    - Anatomia polecenia (82)
    - Napisanie własnego polecenia (83)
- Podsumowanie (85)

Rozdział 7. Projektowanie zaawansowanych aplikacji (87)

- Przejście z prostych funkcji routingu do rozbudowanych kontrolerów (88)
    - Faworyzowanie jawnego routingu (89)
    - Łatwe tworzenie adresów typu REST (89)
- Rozbudowa modeli (90)
    - Sztuczki związane z wydajnością (90)
    - Zabezpieczanie danych miękkimi usunięciami (91)
    - Większa kontrola nad SQL (92)
    - Nasłuchiwanie zdarzeń dotyczących modelu (92)
    - Przydatna klasa paginacji (93)
- Łatwa konfiguracja środowiska (93)
    - Artisan i środowiska (94)
    - Dodawanie własnych ustawień w plikach konfiguracji (95)
- Stosowanie własnych klas (95)
- Wygodna współpraca z kodem po stronie klienta (96)
- Podsumowanie (97)

Dodatek A. Arsenał narzędzi (99)

- Funkcje pomocnicze dla tablic (99)
    - Przykłady użycia funkcji pomocniczych dla tablic (100)
- Obróbka tekstu (101)
    - Funkcje boolowskie (102)
    - Funkcje przekształcające (102)
    - Funkcje odmiany (102)
- Obsługa plików (103)
    - Przesyłanie plików (103)
    - Modyfikacja zawartości plików (104)
- Wysyłanie e-maili (105)
- Biblioteka Carbon, czyli łatwiejsza obsługa daty i czasu (106)
    - Tworzenie obiektów Carbon (106)
    - Wyświetlanie znaczników czasowych przyjaznych dla użytkownika (107)
    - Metody zwracające wartości boolowskie (107)
    - Obsługa Carbon we właściwościach DateTime modeli Eloquent (107)
- Kolejki, czyli wykonywanie długich zadań w tle (108)
    - Utworzenie zadania i wysłanie go do kolejki (108)
    - Nasłuchiwanie kolejki i wykonywanie zadań (109)
    - Informacja o błędzie w wykonaniu zadania (109)
    - Kolejka bez procesu działającego w tle (109)
- Co dalej? (110)

Skorowidz (111)