

Dokumentacja Projektu: CourseMng

2025 Justyna Bokszezanin

Contents

Wprowadzenie.....	3
Specyfikacja funkcjonalna.....	3
Użytkownik.....	3
Koszyk.....	3
Płatność.....	3
Administrator.....	3
Specyfikacja нефunkcjonalna.....	3
Bezpieczeństwo.....	3
Wydajność.....	4
Responsywność.....	4
Skalowalność.....	4
Kompatybilność przeglądarek.....	4
Architektura systemu.....	4
Specyfikacja techniczna.....	6
Struktura projektów.....	6
Opis techniczny rozwiązań.....	7
Mechanizm autoryzacji i autentykacji.....	7
Integracja z systemem Stripe (płatności).....	8
Wymagania systemowe.....	9
Przykładowe widoki aplikacji.....	10

Wprowadzenie

CourseMng to aplikacja typu portal sprzedaży kursów, umożliwiająca użytkownikom przeglądanie i zakup kursów online. Wykładowcy mogą dodawać nowe kursy, edytować je i usuwać, administratorzy przeglądać statystyki a studenci dokonywać zakupów.

Specyfikacja funkcjonalna

Użytkownik

- Rejestracja i logowanie
- Przeglądanie listy kursów
- Dodawanie kursów do koszyka
- Dokonywanie płatności (symulacja)
- Dostęp do zakupionych kursów

Koszyk

- Dodawanie i usuwanie kursów
- Wyświetlanie podsumowania cenowego

Płatność

- Procesowanie płatności
- Historia płatności

Administrator

- Przegląd statystyk dotyczących kursów
- Podgląd płatności

Specyfikacja нефunkcjonalna

Bezpieczeństwo

JWT do autoryzacji, ochrona endpointów.

Wydajność

REST API zoptymalizowane do pobierania tylko niezbędnych danych.

Responsywność

Interfejs dostosowany do urządzeń mobilnych.

Skalowalność

Warstwa serwerowa gotowa do uruchomienia w kontenerach.

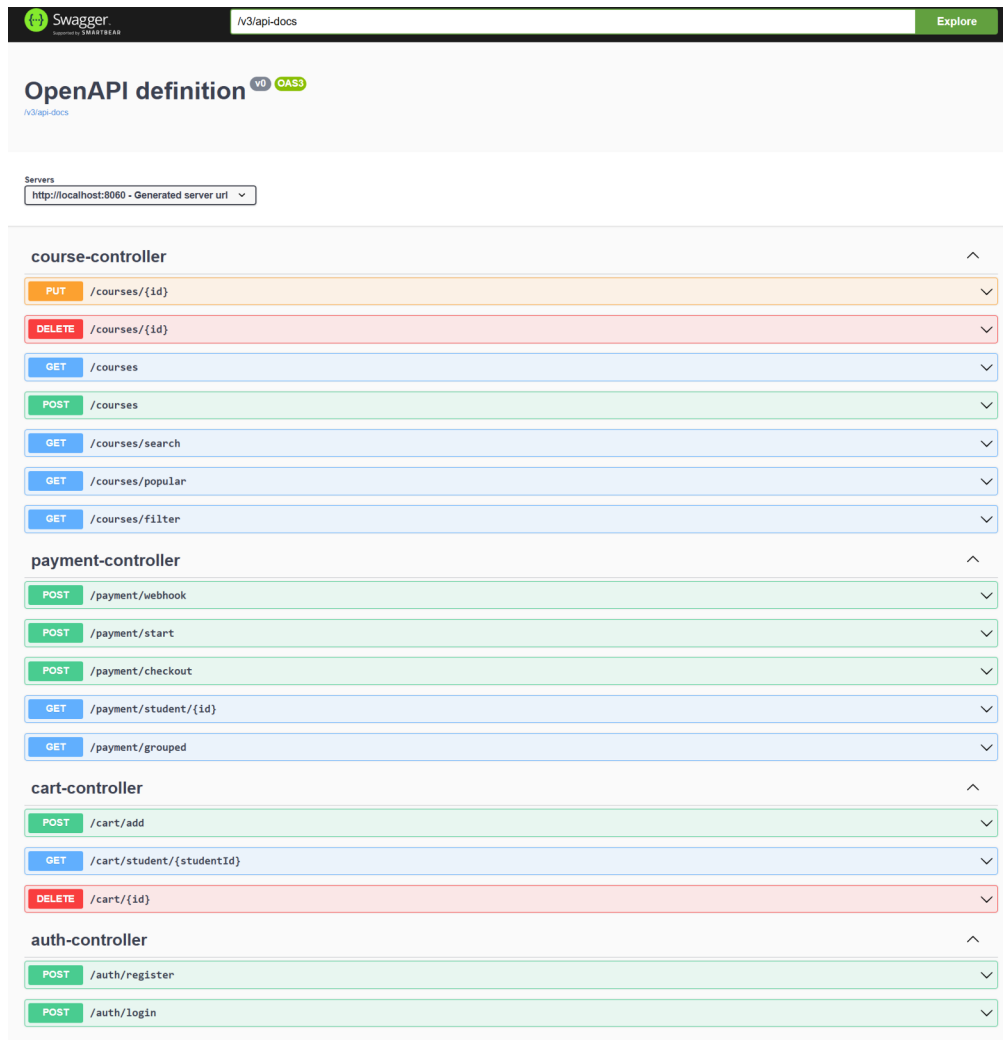
Kompatybilność przeglądarek

Wsparcie dla najnowszych wersji Chrome, Firefox, Edge.

Architektura systemu

Technologie:

- Backend: Spring Boot + JPA + H2/MySQL
- Frontend: React (JavaScript)
- Komunikacja: REST API



Rysunek 1 Widok na dokumentację API

- Autoryzacja i autentykacja: JWT

Witaj w panelu: Student

Wyloguj

Dostępne kursy

Nazwa kursu
Min cena
Max cena
Filtruj

Kurs Python: AI/ML
Zdalny, 4x w tygodniu, 8 tygodni
Cena: 12000.00 PLN
Dodaj do koszyka

Kurs Java: zaawansowany
Zdalny, 3x w tygodniu, 6 tygodni
Cena: Brak ceny
Dodaj do koszyka

Kurs Python: podstawy
Stacjonarny, 2x w tygodniu, 12 tygodni
Cena: Brak ceny
Dodaj do koszyka

Kurs Java: podstawy
Stacjonarny, 1x w tygodniu, 20 tygodni
Cena: 7000.00 PLN
Dodaj do koszyka

Twój koszyk

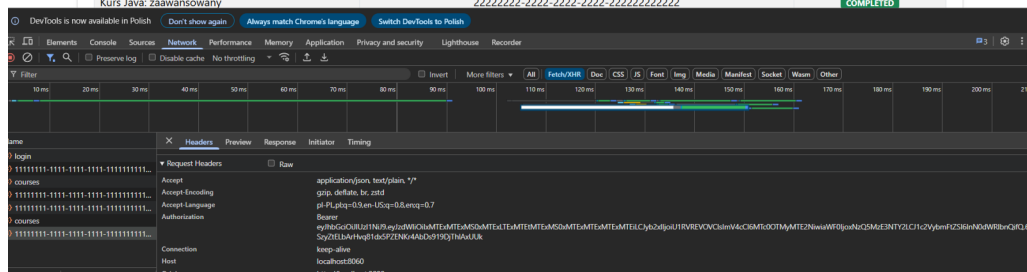
Kurs Python: AI/ML

Opłać Usun

Suma: 12000.00 PLN

Opłacone kursy

Nazwa kursu	ID kursu	Status
Kurs Java: zaawansowany	22222222-2222-2222-2222-222222222222	COMPLETED



Rysunek 2 Token JWT w narzędziu deweloperskim

Specyfikacja techniczna

Struktura projektów

Backend:

- Główne pakiety:
 - controller – kontrolery REST
 - repository – interfejsy JPA
 - dto – obiekty transferowe
 - model – encje JPA
- Endpointy REST:
 - /api/auth/* – rejestracja, logowanie

- /api/courses/* – zarządzanie kursami
- /api/cart/* – koszyk
- /api/payment/* – płatności

Frontend:

- src/api – logika API
- src/components – komponenty widokowe
- src/pages – strony
- src/context – zarządzanie stanem (React Context)

Opis techniczny rozwiązań

Mechanizm autoryzacji i autentykacji

System korzysta z mechanizmu JWT (JSON Web Token) do realizacji autentykacji i autoryzacji użytkowników. Proces wygląda następująco:

Proces logowania:

1. Użytkownik przesyła login i hasło na endpoint /api/auth/login.
2. Backend weryfikuje dane i generuje token JWT z zakodowanymi danymi użytkownika (ID, rola, itd.).
3. Token jest zwracany do klienta i przechowywany w pamięci lokalnej lub kontekście aplikacji frontendowej.

Token JWT zawiera:

- sub – identyfikator użytkownika (UUID)
- role – rola użytkownika (STUDENT, LECTURER, ADMIN)
- exp – data wygaśnięcia tokena

Dostęp do zasobów:

- Każde żądanie do chronionego endpointa musi zawierać nagłówek:
`Authorization: Bearer <token>`
- Token jest weryfikowany przez Spring Security JWT filter.
- Dostęp do endpointów jest zabezpieczony adnotacją:
`@PreAuthorize("hasRole('ADMIN')")`

Role:

- STUDENT – przegląda, kupuje kursy
- LECTURER – tworzy i edytuje kursy
- ADMIN – zarządza statystykami

Integracja z systemem Stripe (płatności)

Aplikacja korzysta z **Stripe Checkout API** do obsługi płatności online. Założone zostało konto na Stripe oraz wygenerowany klucz „Secret Key”. Celem obsługi odpowiedzi zwrotnej włączono lokalnie webhook:

```
C:\>stripe listen --forward-to localhost:8060/payment/webhook --events checkout.session.completed
> Ready! You are using Stripe API Version [2025-05-28.basil]. Your webhook signing secret is whsec_2dfa0f15e8c56de2f6276
030279fd32ddd3a1f2ff96018d534f87dbef7c61089 (^C to quit)
2025-06-07 15:26:02 --> checkout.session.completed [evt_1RXMl3Fh71ZWNWsJOEdNL4ue]
2025-06-07 15:26:02 <-- [200] POST http://localhost:8060/payment/webhook [evt_1RXMl3Fh71ZWNWsJOEdNL4ue]
2025-06-07 16:28:28 --> checkout.session.completed [evt_1RXNjTFh71ZWNWsJTzt5805o]
2025-06-07 16:28:31 <-- [200] POST http://localhost:8060/payment/webhook [evt_1RXNjTFh71ZWNWsJTzt5805o]
```

Rysunek 3 Webhook Stripe

Proces płatności:

1. Użytkownik klika „Opłać” kurs w panelu studenta.
2. Frontend wysyła żądanie POST /payment/start z:

```
{
  "studentId": "UUID",
  "courseId": "UUID",
  "courseName": "Java 101"
}
```
3. Backend tworzy sesję płatności Stripe (SessionCreateParams), określając kwotę i nazwę kursu.
4. Stripe generuje link checkout_url, który jest zwracany do frontendu.
5. Użytkownik zostaje przekierowany na stronę płatności Stripe.

← TEST MODE

Kurs Java: podstawy
7000,00 zł

Lub

E-mail
email@example.com

Metoda płatności

☐ BLIK

☐ Karta

☐ Klarna

☐ Bezpiecznie zapisuj moje dane w celu finalizacji jednym kliknięciem
Płać szybciej w tej witrynie i wszędzie tam, gdzie akceptowana jest usługa Link.

Zapłać

Obsługiwane przez stripe
[Warunki](#) [Prywatność](#)

Rysunek 4 Strona przekierowania Stripe

Potwierdzenie płatności:

- Stripe wysyła webhook na `/payment/webhook` po zakończeniu transakcji.
- Backend przetwarza event Stripe (np. `checkout.session.completed`) i:
 - zapisuje nową płatność (Payment)
 - przypisuje kurs do użytkownika

Inne informacje:

- Stripe obsługuje walutę PLN
- Sesje mają ograniczony czas życia
- Obsługiwane statusy płatności: PENDING, COMPLETED, FAILED

Sprawdzanie świąt państwowych (Holiday API)

Aplikacja zawiera zintegrowany mechanizm sprawdzania, czy aktualny dzień jest świętem państwowym w Polsce. Informacja ta wykorzystywana jest m.in. do wyświetlania komunikatu w stopce aplikacji frontendowej.

Endpoint REST

- **URL:** `/holiday`
- **Metoda:** GET

- **Typ odpowiedzi:** application/json
- **Opis:**
Endpoint pobiera z zewnętrznego API dane o świętach państwowych w Polsce i sprawdza, czy dzisiejsza data odpowiada któremukolwiek z nich.

Źródło danych:

- Zewnętrzne API: [Nager.Date Public Holidays API](https://date.nager.at/api/v3/PublicHolidays/2025/PL)
- Przykładowy request: GET
`https://date.nager.at/api/v3/PublicHolidays/2025/PL`

Wymagania systemowe

Backend:

- Java 17+
- Maven
- Baza danych (PostgreSQL)

Frontend:

- Node.js 18+
- npm lub yarn

Przykładowe widoki aplikacji

- Widok główny listy kursów, koszyka oraz płatności

Witaj w panelu: Student

Wyloguj

Dostępne kursy

Nazwa kursu

Min cena

Max cena

Filtruj

Kurs Python: AI/ML

Zdalny, 4x w tygodniu, 8 tygodni

Cena: 12000.00 PLN

Dodaj do koszyka

Kurs Java: zaawansowany

Zdalny, 3x w tygodniu, 6 tygodni

Cena: Brak ceny

Dodaj do koszyka

Kurs Python: podstawy

Stacjonarny, 2x w tygodniu, 12 tygodni

Cena: Brak ceny

Dodaj do koszyka

Kurs Java: podstawy

Stacjonarny, 1x w tygodniu, 20 tygodni

Cena: 7000.00 PLN

Dodaj do koszyka

Twój koszyk

Kurs Python: AI/ML

Oplac

Usuń

Suma: 12000.00 PLN

Opłacone kursy

Nazwa kursu	ID kursu	Status
Kurs Java: zaawansowany	22222222-2222-2222-2222-222222222222	COMPLETED
Kurs Python: podstawy	33333333-3333-3333-3333-333333333333	COMPLETED
Kurs Java: podstawy	11111111-1111-1111-1111-111111111111	COMPLETED
Suma		7000.00 PLN

Rysunek 5 Widok studenta

- Widok wykładowcy

Witaj w panelu: Lecturer

Wyloguj

Dodaj kurs

Nazwa

Opis

Cena w PLN (np. 50.00)

Dodaj

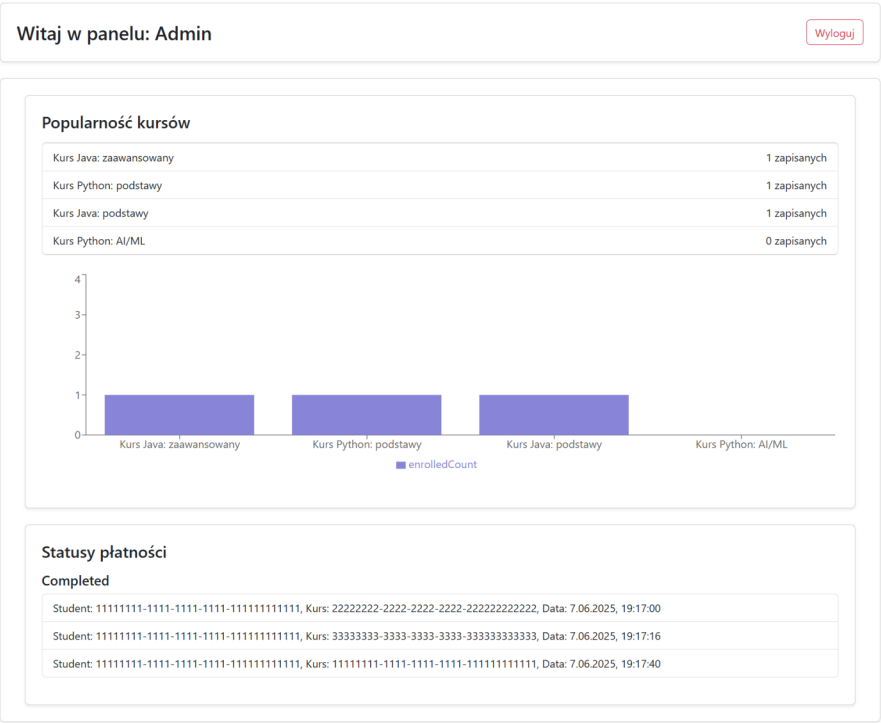
Moje kursy

Filtruj po nazwie...

Nazwa	Opis	Cena [PLN]	Akcje	
Kurs Python: AI/ML	Zdalny, 4x w tygodniu, 8 tygodni	12000.00	Edytuj	Usuń
Kurs Java: zaawansowany	Zdalny, 3x w tygodniu, 6 tygodni	NaN	Edytuj	Usuń
Kurs Python: podstawy	Stacjonarny, 2x w tygodniu, 12 tygodni	NaN	Edytuj	Usuń
Kurs Java: podstawy	Stacjonarny, 1x w tygodniu, 20 tygodni	7000.00	Edytuj	Usuń

Rysunek 6 Widok wykładowcy

- Widok administratora



Rysunek 7 Widok administracyjny

- Ekran logowania/rejestracji

Logowanie

Nazwa użytkownika

Hasło

Zaloguj się

Zarejestruj się

Rysunek 8 Widok logowania

The image shows a registration form titled "Rejestracja" (Registration) centered on a light gray background. The form is a white card with rounded corners. It contains the following elements from top to bottom: a title "Rejestracja" in bold black font; a label "Nazwa użytkownika" (Username) above a text input field; a label "E-mail" above a text input field; a label "Hasło" (Password) above a text input field; a label "Rola" (Role) above a dropdown menu with the placeholder text "Wybierz rolę" and a downward arrow; and a blue button with the white text "Zarejestruj się" (Register).

Rysunek 9 Widok rejestracji

- Koszyk i płatność

←  TEST MODE

Kurs Python: AI/ML

12 000,00 zł

Zapłać z  link

Lub

E-mail

test@test.com

Metoda płatności

☐  BLIK

☒  Karta

Informacje o karcie

4242 4242 4242 4242 

12 / 34

111 

Imię i nazwisko posiadacza karty

test

Kraj lub region

Polska 

☐  Klarna

☐ Bezpiecznie zapisuj moje dane w celu finalizacji jednym kliknięciem

Plać szybciej w tej witrynie i wszędzie tam, gdzie akceptowana jest usługa Link.

Zapłać

Obsługiwane przez 

[Warunki](#) [Prywatność](#)

Rysunek 10 Widok na płatność za kurs poprzez Stripe