

UNIWERSYTET WARMIŃSKO MAZURSKI W OLSZTYNIE

WYDZIAŁ MATEMATYKI I INFORMATYKI

IGOR KAWCZYŃSKI

Kierunek: Informatyka

**Aplikacja webowa wspomagająca przeprowadzanie kursów na żeglarstwo
wodne i rezerwację rejsów – panel klienta.**

Praca inżynierska

**wykonana w Katedrze Metod
Matematycznych Informatyki**

pod kierunkiem dr Pawła Drozdy

Olsztyn 2024

UNIVERSITY OF WARMIA AND MAZURY IN OLSZTYN
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

IGOR KAWCZYŃSKI

Field of study: Computer Science

**Web application supporting the conduct of water sailing courses and
booking cruises – customer panel**

Engineering Thesis

**Written in Chair of Mathematical
Methods of Informatics**

under supervision of dr Paweł Drozda

Olsztyn 2024

Spis treści

Spis rozdziałów

Streszczenie	5
Streszczenie w języku polskim.....	5
Abstract.....	5
1.Wstęp.....	7
1.1 Wprowadzenie	7
1.1 Cel i zakres pracy dyplomowej	8
2. Wykorzystane technologie	10
3. Specyfikacja systemu	18
3.1 Opis analityczny	18
3.2 Lista funkcjonalności (Przypadki użycia)	18
3.3 Grupy użytkowników	19
3.4 Diagramy przypadków użycia	21
3.5 Scenariusze	24
4. Projekt bazy danych	40
4.1 Schemat bazy danych	40
4.2 Opis szczegółowy encji i relacji bazy danych	41
5. Implementacja	47
5.1 Architektura backendu.....	47
5.2 Architektura frontendu.....	55
5.3 Projektowanie logotypu	57
6. Instrukcja użytkownika aplikacji.....	59
6.1 Strona główna	59
6.2 Zakładka Oferta	62
6.3 Zakładka O firmie	66
6.4 Rejestracja.....	68

6.5 Zapis na kurs	70
6.6 Moje kursy	72
6.7 E-learning	73
6.8 Moje eventy	77
6.9 Mój profil.....	78
6.10 Strona 404.....	79
7. Testy	80
7.1 Testy backendowe	80
7.2 Testy frontendowe	82
7.3 Testy osób trzecich	82
8. Podsumowanie.....	83
8.1 Wnioski.....	83
8.2 Możliwości rozwoju	83
Bibliografia.....	85
Literatura	85
Dokumentacja wykorzystanych technologii.....	86
Spis diagramów	87
Spis tabel	88
Spis ilustracji	89

Streszczenie

Streszczenie w języku polskim

Głównym zadaniem pracy dyplomowej jest usprawnienie oraz ułatwienie nauki i zapisów na kursy żeglarskie oraz rezerwacji na rejsy – wszystko przy pomocy stworzonej aplikacji webowej.

Strona umożliwia przegląd nadchodzących kursów i eventów oraz zapis na konkretny kurs dla zalogowanego klienta oraz rezerwację na rejs przy pomocy wypełnionego formularza. Użytkownik, który opłacił kurs, ma możliwość korzystania z nauki (e-learning) w celu przygotowania się do egzaminu kończącego kurs. E-learning pozwala na wylosowanie pytania z konkretnego działu oraz sprawdzenie poprawnej odpowiedzi, wygenerowanie egzaminu próbnego, lub przejrzenie statystyk nauki.

Oprócz głównych założeń, aplikacja pozwala nam również przejrzeć zbiór informacji o firmie świadczącej usługi oraz odpowiedzi na najczęściej zadawane pytania w formie FAQ. Do dyspozycji na stronie znajduje się także cennik z ofertą żeglarską, gdzie widoczne są poszczególne terminy kursów lub rejsów wraz z zapisaną liczbą uczestników. Cenną funkcjonalnością strony jest również profil klienta, gdzie możemy edytować swoje dane dla konta, bądź sprawdzić przypisane dla nas kursy lub rejsy i nimi zarządzać.

Abstract

The primary objective of the dissertation is to enhance and simplify the process of learning, enrolling in sailing courses, and booking voyages, all through a specially developed web application.

This site allows users to view upcoming courses and events, and enables registered clients to sign up for a specific course or reserve a spot in an event via form. Users who have paid for a course gain access to e-learning resources, designed to help them prepare for the final examination. The e-learning system offers functionalities such as selecting questions from specific categories, verifying correct answers, generating mock exams, and check learning statistics.

In addition to its core features, the application also provides comprehensive information about the service-providing company and a FAQ section for commonly asked questions. The website also includes a detailed price list showcasing various course offerings, along with

their scheduled dates and the current number of registered participants. Another valuable aspect of the site is the client profile section, where users can edit their account details, and manage their enrolled courses or booked voyages.

1. Wstęp

1.1 Wprowadzenie

W erze cyfryzacji, wiele branż doświadcza rewolucji w sposobie zarządzania i organizacji pracy. Pomimo tego, wiele jednostek i instytucji w Polsce, zajmujących się organizacją rejsów i kursów żeglarskich, nadal bazuje na tradycyjnych metodach, przede wszystkim opierając swój biznes na papierowej kartce i komunikacji telefonicznej.

W kontekście żeglarstwa, sektorze z bogatą historią, wprowadzenie nowoczesnej technologii umożliwiającej szybkie i wygodne zapisy na kursy bądź rejsy żeglarskie w połączeniu z możliwością nauki do uzyskania konkretnego patentu żeglarskiego, może znaczowo wpływać na popularność tej branży i usprawnić przeprowadzanie tych procesów.

Niniejsza praca skupia się na tworzeniu aplikacji webowej, która usprawni wymienione procesy i zaoferuje szybki oraz wygodny interfejs, który otworzy nowe możliwości dla klientów oraz organizatorów, jednocześnie podnosząc standardy obsługi i efektywność zarządzania.

Ponadto, aplikacja dzięki wprowadzeniu interaktywnych elementów e-learningu dla kilkunastu rodzajów kursów, jest w stanie znacznie wpływać na pozytywne wyniki uzyskiwane w egzaminach kończących, niezbędnych do uzyskania konkretnego patentu żeglarskiego. Implementując łatwy dostęp do praktycznych narzędzi naukowych, aplikacja ta ma potencjał, aby w większym stopniu zniwelować barierę wejścia dla nowych entuzjastów żeglarstwa i możliwe wpływać na popularność tego hobby lub profesji.

Przez zaprojektowanie i wdrożenie innowacyjnych rozwiązań, niniejsza praca ma na celu nie tylko przedstawienie funkcjonalnego narzędzia online, ale także zobrazowanie, w jaki sposób nowoczesna aplikacja webowa jest w stanie przyczynić się do rozwoju i ulepszenia konkretnych sektorów życia.

Ważnym aspektem aplikacji była ścisła współpraca i liczne konsultacje z właścicielem firmy zajmującej się wyżej wymienioną organizacją kursów i rejsów. To pozwoliło na większe zrozumienie wszelkich wymagań i oczekiwania zarówno potencjalnego klienta, jak i organizatora. Dzięki tej współpracy, projekt aplikacji był stale modyfikowany i dopasowywany tak, aby zapewnić, że finalny produkt będzie w stanie spełnić oczekiwania obu stron.

Realizacja tego projektu stanowiła cenną okazję do zastosowania i rozwinięcia zaawansowanych umiejętności w dziedzinie projektowania, implementacji i testowania

oprogramowania. Praca ta wymagała dogłębnej analizy i wykorzystania nowoczesnych technik programowania oraz zwrócenia na aspekty takie jak wydajność, skalowalność i utrzymanie dobrej jakości kodu. Przez zastosowanie standardów clean code oraz wykorzystanie nowych narzędzi, projekt ten nie tylko przyczynił się do optymalizacji procesów zapisu na kursy bądź rezerwacji rejsów żeglarskich, ale także stanowił platformę do nagłego rozwoju zawodowego, przyczyniając się do pogłębienia wiedzy technicznej i praktycznych umiejętności w zakresie inżynierii oprogramowania.

1.1 Cel i zakres pracy dyplomowej

Celem omawianej pracy dyplomowej jest przede wszystkim zaprojektowanie oraz wdrożenie oprogramowania aplikacji internetowej, która będzie w stanie wydajnie i bezbłędnie funkcjonować, dostarczając przy tym szereg możliwości w obrębie kursów i rejsów żeglarskich oraz przedstawienie procesu tworzenia omawianej aplikacji. W ramach projektu powstał także panel administracyjny pozwalający na zarządzanie bazą danych, którym zajął się Pan Grzegorz Krych, w niniejszej pracy priorytetem będzie jednak część kliencka, którą opracowałem ja, Igor Kawczyński. Kluczowym elementem jest omówienie funkcjonalności, jakie dostarcza aplikacja, w tym zarówno te bezpośrednio widoczne dla użytkownika, jak i te działające w tle, zapewniające płynność i efektywność systemu. Praca dostarcza również informacji o analitycznym aspekcie aplikacji, czyli omawia uwzględnione w niej scenariusze, przypadki użycia, diagramy, a także sposób implementacji, czyli przedstawienie wykorzystanych technologii i narzędzi, które zostały użyte do stworzenia oprogramowania, wraz z uzasadnieniem ich wyboru i opisem, w jaki sposób przyczyniły się one do realizacji założonych celów projektowych.

Zakres głównych zadań :

- Projekt relacyjnej bazy danych aplikacji
- Utworzenie dokumentu analitycznego uwzględniając podejście UML
- Utworzenie listy zadań z podziałem pracy na panel administratora i panel klienta
- Sporządzenie opisu funkcjonalności aplikacji
- Wykonanie logotypu aplikacji
- Zaprojektowanie szablonów widoków stron aplikacji
- Implementacja aplikacji od strony backendowej i frontendowej
- Przeprowadzenie testów jednostkowych

Warto dodatkowo nadmienić, że aplikacja powstała w sposób umożliwiający dalszy jej rozwój, który mógłby przyczynić się do wykorzystania jej w szerszym spektrum w zakresie branży żeglarskiej.

2. Wykorzystane technologie

Aplikacje webowe w ostatnich latach zyskały w dużym stopniu na popularności, przede wszystkim ze względu na łatwą i uniwersalną dostępność, gdzie możemy z poziomu przeglądarki na naszych urządzeniach mobilnych (smartfon, tablet) lub urządzeniach stacjonarnych (laptop, komputer osobisty), przejść na konkretną stronę i korzystać z jej funkcjonalności. Takie rozwiązanie ma zdecydowaną przewagę nad aplikacjami desktopowymi, które musimy instalować na naszym sprzęcie i używać przede wszystkim własnej mocy obliczeniowej do korzystania z ich funkcjonalności – w aplikacji webowej natomiast większe znaczenie ma sam serwer, który daną aplikację utrzymuje. Pełnoprawne aplikacje webowe składają się z trzech kluczowych dla nich części, frontendu, backendu oraz bazy danych.

Baza danych:

- **Przechowuje** – Baza danych przechowuje całe struktury danych, najczęściej w postaci tabel i odpowiadających im rekordów o danym typie
- **Integruje** – Baza danych odpowiada także za utrzymanie bazy spójnej i poprawnej, chodzi konkretnie o sprawdzanie ograniczeń spowodowanych kluczami obcymi lub specjalistycznymi typami danych
- **Operuje** – Baza danych jest w stanie wykonywać zapytania, użytkownik dzięki temu jest w stanie przeprowadzać operacje CRUD (Create, Read, Update, Delete), które są kluczowe do przetwarzania wszelkich informacji

Backend:

- **Przetwarzanie wszelkich żądań użytkownika** – Backend odbiera wysłane żądania HTTP przesłane przez frontend lub innych klientów i przetwarza je według napisanej logiki systemu
- **Zarządzanie bazą danych** – Backend jest odpowiedzialny za komunikację pomiędzy logiką systemu, a bazą danych, dzięki czemu jesteśmy w stanie tworzyć, pobierać, aktualizować oraz usuwać dane w niej przechowywane. To zazwyczaj zapytania napisane w języku SQL lub jego pochodne

- **Logika biznesowa** – Backend obejmuje całą logikę biznesową aplikacji internetowej, która nakreśla jak cały system ma działać. To wszelkie reguły, algorytmy i wyliczenia niezbędne do korzystania z funkcjonalności aplikacji
- **Udostępnienie API (interfejsu programowania aplikacji)** – Backend umożliwia frontendowi lub innym klientom komunikację z logiką aplikacji, najczęściej w formie REST API, GraphQL lub WebSocket
- **Bezpieczeństwo** – Backend powinien obsłużyć wszelkie sytuacje zagrażające bezpieczeństwu danych, między innymi są to ataki typu Cross Site Request Forgery, Cross Site Scripting czy SQL Injection
- **Wydajność** – Backend dba o szybkość działania aplikacji i tempo w jakim przetwarzane są wszelkie informacje w jej obrębie

Frontend:

- **Wygląd i układ interfejsu strony** – Frontend odpowiada za generowanie widoków, czyli między innymi pasek nawigacyjny, nagłówki, tekst, przyciski, obrazy i ich stylowanie, najlepiej w sposób responsywny, czyli taki, który obsługuje ekrany o różnych proporcjach i rozmiarach, chodzi przede wszystkim o urządzenia mobilne (mniejsze ekrany), a także komputery stacjonarne i ich większe wyświetlacze
- **Interakcje z aplikacją** – Frontend zapewnia możliwość interakcji klienta z całą aplikacją, mianowicie obsługą myszki i klawiatury
- **Wysyłanie żądań do backendu** – Frontend odpowiada także za wysyłanie wyżej wymienionych żądań do serwera backendowego, który obsługuje je dzięki swojej logice biznesowej

Istnieją oczywiście odstępstwa od tych reguł, bywa nawet tak, że to na frontendzie znajduje się lwna część walidacji, którą także można nazywać logiką systemu. Wszystko jest zależne od rodzaju i przeznaczenia konkretnej aplikacji internetowej. W przypadku aplikacji stworzonej w celach pracy dyplomowej, zostały zachowane jednak główne konwencje integracji frontenu z backendem.

Do zbudowania wszystkich trzech komponentów aplikacji użyte zostały technologie takie jak:

Baza danych:

- **PostgreSQL** - Jako język zarządzania bazą danych, wybrano PostgreSQL ze względu na jego wydajność, niezawodność, wsparcie dla dużych zbiorów danych i wygodną aplikację w postaci PgAdmin – wersja 15
- **Flyway** – Narzędzie do zarządzania migracjami baz danych. Pozwala na uspójnienie skryptów i ich automatycznewdrażanie. Szczególnie przydatny w przypadku implementacji kodu dla większej liczby osób – stąd został użyty w aplikacji wymienionej w pracy dyplomowej, aby skrypty z panelu administratora nie konfliktovaly z tymi od strony klienta i na odwrót – wersja 9

Technologie Backendowe:

- **Java** – Główny język programistyczny wykorzystany w aplikacji. Wysokopoziomowy, kompilowany, silnie obiektowy i z restrykcyjną kontrolą typów, gdzie składnia wzorowana jest na C++. Java razem z Pythonem jest najczęściej używanym językiem do tworzenia nowoczesnych systemów aplikacji webowych – stąd też wybór języka nie był szeroki i padł właśnie na Javę, która względem Pythona charakteryzuje się kompilacją do kodu bajtowego, co korzystnie wpływa na wydajność i wymienioną wcześniej silną kontrolą typów, co wpływa równie korzystnie na intuicyjność pisanej kodu. Java użytkowana jest także do tworzenia oprogramowania desktopowego czy gier. Ponadto, system operacyjny Android działa w środowisku Javy, co pozwala na intuicyjne tworzenie aplikacji lub gier na Androida. Jest preferowanym językiem w wielu korporacjach i dużych firmach. Często wykorzystywana w systemach bankowych i ubezpieczeniowych, ze względu na oferowane przez ten język solidne rozwiązania w zakresie zabezpieczeń. Wersja Javy użyta w projekcie to 17
- **Spring** – Framework¹ dla języka Javy, stworzony z myślą o implementowaniu aplikacji wysokiej jakości, które są bezpieczne, skalowalne i stosunkowo łatwe do zarządzania. Spring oferuje mnóstwo dodatków i narzędzi, które ułatwiają rozwijanie

¹ Framework – zestaw narzędzi, bibliotek i konwencji, które przyspieszają proces wytwarzania oprogramowania

wielu aspektów aplikacji, takich jak obsługa bazy danych, obsługa transakcji, wykorzystanie REST, czy ich zabezpieczenie i testowanie. Jedną z głównych zalet Springa jest inwersja kontroli, która umożliwia oddzielenie zależności i konfiguracji od rzeczywistego kodu biznesowego. Dzięki temu kod staje się łatwiejszy w utrzymaniu i testowaniu. Spring jest także znany ze swojego potężnego mechanizmu wstrzykiwania zależności (Dependency Injection), który pozwala na wygodne zarządzanie komponentami aplikacji i jest swoistą techniką programowania, która została zresztą szeroko użyta w omawianej aplikacji.

- **Spring Boot** – Projekt Springa, którego główną ideą jest zminimalizowanie implementacji licznej, czasami zbędnej konfiguracji, co pozwala na dużo szybszy start dla stworzenia aplikacji względem klasycznego Springa. Oprócz zmian względem konfiguracji, Spring Boot dostarcza także serwer aplikacyjny Tomcat, który obsługuje nasze aplikacje oraz masę innych udogodnień, które pozytywnie wpływają na możliwości i szybkość rozwoju kodu backendowego. Wersja Springa użyta w projekcie to 3.1.5
- **Spring Security** – Wysoce konfigurowalny framework, służący do zabezpieczenia aplikacji. Wspiera nowoczesne protokoły i standardy zabezpieczeń. Użyty przede wszystkim do zabezpieczenia wszelkich żądań HTTP poprzez filtrowanie i autoryzacji z uwierzytelnieniem. Wykorzystany w wersji 6.1.5
- **Java Persistence API** – Specyfikacja dostarczająca Javie sposób mapowania obiektów na rekordy w bazie danych (ORM – Object Relational Mapping), co pozwala na tworzenie obiektów w kodzie, aby następnie przekonwertować je na rekordy pochodzące z konkretnych encji. JPA definiuje zestaw interfejsów, reguł i adnotacji, które można w łatwy sposób wykorzystać, dzięki czemu programista poświęca mniej czasu na komunikację pomiędzy kodem backendowym, a bazą danych i może skupić się na logice biznesowej aplikacji. Wersja JPA użyta w projekcie to 3.1.0
- **Hibernate** – Framework implementujący JPA i dostarczający własne zapytania typu HQL (Hibernate Query Language), a także własny sposób zarządzania sesją i pamięcią cache. Wersja użyta w projekcie to 6.2.13
- **JUnit** – Framework przeznaczony do automatycznego testowania jednostkowego i zarazem narzędzie, które jest standardem de facto pisania i uruchamiania testów w projektach Java. JUnit jest częścią rodziny xUnit i został zaprojektowany w celu

zachęcenia do pisania testów na wczesnym etapie procesu tworzenia oprogramowania oraz promowania idei "testów jako dokumentacji". Użyty w celu przetestowania konkretnych części aplikacji. Wersja to 5.9.3

- **Mockito** – Framework umożliwiający tworzenie symulacji obiektów, czyli tak zwanych „mocków”. Takie sztuczne obiekty pozwalają na odtworzenie prawdziwych obiektów i tym samym rzeczywistych scenariuszy, które mogą wystąpić w samej aplikacji. Użyto w wersji 5.3.1
- **Jakarta Validation API** – Framework zapewniający mechanizm automatycznego walidowania obiektów, oparty na personalizowanych adnotacjach i modyfikowalnych regułach. Wersja frameworka to 3.0.2
- **Lombok** – Biblioteka Javy, która automatyzuje proces pisania stereotypowego, powtarzalnego kodu, takiego jak metody get, set, metody equals(), hashCode() i toString(), a także konstruktory. Jest to narzędzie czasu komplikacji, które za pomocą adnotacji w kodzie źródłowym generuje kod pośredni (bytecode) podczas komplikacji, dodając potrzebne metody do klas bez konieczności ręcznego ich pisania. Jest to krótko mówiąc narzędzie redukujące tak zwany „boilerplate code”. Wersja to 1.18.30
- **JSON Web Token (JWT)** – Kompaktowy sposób na bezpieczne przesyłanie informacji między stronami jako obiekt JSON². Informacje te mogą być zweryfikowane i zaufane, ponieważ są cyfrowo podpisane. JWT jest często używany do uwierzytelniania w aplikacjach internetowych i do przesyłania informacji między serwerem a klientem. Użyty w niniejszym projekcie do zabezpieczenia autoryzacji użytkownika, w wersji 0.11.5

Technologie Frontendowe:

- **HTML** – Fundament każdej strony internetowej, język znaczników używany do tworzenia i strukturyzowania stron internetowych. Służy do określania treści i elementów strony, takich jak tekst, obrazy, wideo, linki i inne. Pozwala na organizację treści, nadając im odpowiednią formę poprzez znaczniki, takie jak paragrafy, listy czy sekcje. Użyty w wersji 5
- **CSS** – Język używany do kontrolowania stylizacji i ogólnych wizualiów tworzonych widoków na stronie internetowej. Umożliwia definiowanie wyglądu takich elementów

² JSON (JavaScript Object Notation) - lekki format wymiany danych oparty na strukturze klucz:wartość, który jest łatwy do czytania i zapisywania dla ludzi oraz łatwy do generowania dla maszyn.

jak czcionka, kolory, rozmiary bloków, odstępy pomiędzy blokami, marginesy i wiele innych. CSS to kluczowy element tworzenia responsywnych i estetycznych aplikacji webowych. Użyty w wersji 3

- **JavaScript** – Język skryptowy tworzący dynamikę i interaktywność aplikacji. Pozwala na obsługę konkretnych zdarzeń na stronie, manipulację danymi, komunikację z serwerem, tworzenie animacji. Użyty w wersji ES2022
- **TypeScript** – Język rozszerzający JavaScript o statyczne typowanie, bardziej zaawansowane funkcje programistyczne i lepsze wsparcie dla środowisk programistycznych. Zdecydowanie ułatwia pisanie złożonych aplikacji ze względu na możliwość definiowania interfejsów i typów danych. Kompiluje się do JavaScriptu, co pozwala na praktycznie nieograniczone wsparcie przeglądarek. Użyty w wersji 4.7.5
- **NPM** – System zarządzania pakietami dla języka JavaScript i zarazem podstawowe narzędzie używane w środowisku Node.js, ale także powszechnie stosowane w zarządzaniu zależnościami dla front-endu. NPM umożliwia programistom łatwe instalowanie, udostępnianie i zarządzanie bibliotekami i modułami JavaScript. Działa w oparciu o repozytorium online, które zawiera tysiące pakietów napisanych przez społeczność JavaScript. NPM pozwala na szybkie dodawanie funkcjonalności do projektów poprzez instalację gotowych modułów, co znacząco przyspiesza rozwój oprogramowania. Jest on także narzędziem do publikowania własnych bibliotek, dzięki czemu inni programiści mogą z nich korzystać. Użyto w wersji 9.6.7
- **Angular** – Framework opracowany przez Google, używany do implementowania dynamicznych aplikacji internetowych. Oferuje bogaty zestaw funkcji począwszy od zabezpieczonych przekierowań do dwustronnego wiązania danych. Angular umożliwia tworzenie aplikacji, które działają płynnie zarówno na urządzeniach mobilnych, jak i desktopowych, co jest kluczowe w dzisiejszym świecie internetu. Jego modularna natura pozwala na łatwe zarządzanie dużymi projektami, a system szablonów oraz wbudowane funkcje wspierają tworzenie interaktywnych interfejsów użytkownika. Obok frameworka React najpopularniejszy w kontekście tworzenia aplikacji webowych. Wybrany ze względu na dużą intuicyjność i spore podobieństwo do struktury projektu pisanej w Springu. Użyty w wersji 16
- **Angular CLI** – Narzędzie rozszerzające framework Angular o linię poleceń pozwalającą na szybkie tworzenie, modyfikację i testowanie komponentów frontendowych. Używana wersja w projekcie to 16.2.10

- **Jasmine** – Framework umożliwiający pisanie testów jednostkowych po stronie frontendowej w czytelny, jasny sposób. Użyty w wersji 4.6.0
- **Bootstrap** – Framework do tworzenia nowoczesnych i skalowalnych stylów stron internetowych. Znaczco ułatwia projektowanie interfejsu użytkownika dzięki gotowym komponentom i klasom stylowym. Używany w wersji 5.3.2
- **PrimeNG** – Kolekcja gotowych komponentów dostarczających interfejs i funkcjonalności, które są dostosowane do pracy z frameworkm Angular. Biblioteka ta oferuje szeroki wybór elementów interfejsu, takich jak przyciski, pola formularzy, panele danych, menu i wiele innych. Każdy z komponentów PrimeNG jest zaprojektowany z myślą o integracji z architekturą Angulara, co umożliwia łatwe włączanie ich do aplikacji oraz ich dostosowywanie. PrimeNG wyróżnia się estetycznym designem i wysoką konfigurowalnością. Komponenty są stylizowane i gotowe do użycia, jednocześnie oferując dużą elastyczność w kwestii dostosowania wyglądu do specyficznych potrzeb projektu. Biblioteka ta jest również znana z obsługi zaawansowanych funkcji, takich jak obsługa responsywności, wsparcie dla touch interfejsów oraz dostępności. Wersja to 16.1
- **Angular Material** – Podobnie jak PrimeNG, narzędzie dostarczające komponenty, które umożliwiają przyspieszenie tworzenia widoków na stronie. Dedykowana kolekcja stylów dla frameworka Angular, stworzona przez Google. Używana wersja 16.2.12

Inne:

- **Postman** - Narzędzie do testowania API (Application Programming Interface). W kontekście programowania omawianej aplikacji typu REST w Spring Boot, Postman umożliwił wysyłanie żądań HTTP do serwera aplikacji i odbieranie odpowiedzi, co było kluczowe w debugowaniu i testowaniu wszelkich funkcjonalności
- **Trello** – Aplikacja pomocna w zarządzaniu projektem zespołowym poprzez tworzenie listy tablic zawierających zadania do wykonania. Użyta w celu podziału prac dla panelu administratora i panelu klienta i dla śledzenia wszelkich zmian w projekcie aplikacji
- **Figma** – Nowoczesne narzędzie do projektowania interfejsów użytkownika i jego doświadczenia (UI/UX), które zyskało popularność dzięki swoim zaawansowanym funkcjom i możliwościami współpracy w zespole, co było szczególnie przydatne dla

niniejszej pracy ze względu na równoległe prowadzenie panelu administratora i klienta. Jest to aplikacja oparta na przeglądarce, co oznacza, że nie wymaga instalacji oprogramowania na komputerze użytkownika, podobnie jak projekt do pracy dyplomowej. W Figmie zaprojektowany został cały interfejs graficzny aplikacji

- **Photoshop** - Oprogramowanie do tworzenia, modyfikowania i manipulowania zdjęciami, grafikami i innymi rodzajami obrazów. Program ten został stworzony przez firmę Adobe i jest jednym z najbardziej popularnych narzędzi do obróbki grafiki na świecie. Wykorzystany do stworzenia logotypu aplikacji w wersji dla roku 2021

3. Specyfikacja systemu

3.1 Opis analityczny

Celem opracowanej aplikacji webowej było przede wszystkim usprawnienie procesu zapisów na kursy żeglarskie oraz rezerwacji na rejsy. System ten, zaprojektowany z myślą o intuicyjnym użytkowaniu, umożliwia użytkownikom łatwe przeglądanie dostępnych kursów i rejsów, wraz z ich terminami i cenami, co znaczco ułatwia proces wyboru odpowiedniej oferty. Sam wybór oferty polega na wybraniu konkretnego rodzaju kursu bądź rejsu, a następnie terminu i miejsca wydarzenia. Warto dodatkowo podkreślić, że zapis na kursy odbywa się w kontekście konta klienta, czyli konkretny kurs przypisany jest do konkretnego użytkownika, rejsy natomiast rezerwujemy na podany w formularzu e-mail, co pozwala na rezerwację bez posiadania konta użytkownika.

Kluczowym aspektem aplikacji jest również wprowadzenie modułu e-learningowego, który pozwala użytkownikom na zdalne przygotowanie się do egzaminu na patent żeglarski. Ten segment aplikacji zawiera funkcje umożliwiające dostęp do losowo generowanych pytań z konkretnego działu żeglarskiego, generowanie egzaminów próbnych oraz analizę postępów w nauce, co stanowi istotne wsparcie w procesie edukacyjnym.

Aplikacja została także wyposażona w funkcje logowania i rejestracji, co zapewnia bezpieczne i personalizowane środowisko dla każdego użytkownika. Dodatkowo, z myślą o wsparciu użytkowników i ułatwieniu dostępu do ważnych informacji, system zawiera sekcję FAQ, odpowiadającą na najczęściej zadawane pytania związane z rejsami, kursami oraz procesem nauczania żeglarskiego, a także sekcję kontaktową, która może być wykorzystywana przez firmę świadczącą usługi żeglarskie do komunikacji z potencjalnymi klientami. Oprócz tego, aplikacja oferuje możliwość zarządzania profilem użytkownika, co obejmuje edycję danych osobowych oraz przeglądanie statystyk aktywności w ramach systemu.

3.2 Lista funkcjonalności (Przypadki użycia)

- Rejestracja konta użytkownika
- Logowanie
- Wylogowanie
- Wyświetlenie statystyk profilu

- Edycja profilu
- Wyświetlenie dostępnych kursów
- Wyświetlenie dostępnych rejsów
- Nauka ogólna (wygenerowanie losowego pytania)
- Wygenerowanie egzaminu próbnego
- Rozwiążanie egzaminu próbnego
- Podgląd zbliżających się kursów użytkownika
- Wycofanie się z kursu
- Podgląd zbliżających się rejsów użytkownika
- Wycofanie się z rejsu
- Wyświetlenie informacji kontaktowych
- Wyświetlenie najczęściej zadawanych pytań o kursy i rejsy

3.3 Grupy użytkowników

W ramach opracowanego systemu, zdefiniowane zostały trzy główne grupy użytkowników, każda z różnym poziomem dostępu do głównych funkcjonalności systemu. Hierarchia dostępu i uprawnień jest kluczowym elementem zapewniającym efektywność i bezpieczeństwo operacji w systemie.

Dostępy	Gość ³	Użytkownik zalogowany	Administrator
Strona główna	TAK	TAK	TAK
Rejestracja konta użytkownika	TAK	TAK	TAK
Logowanie	NIE	TAK	TAK
Wylogowanie	NIE	TAK	TAK
Wyświetlenie dostępnych kursów	TAK	TAK	TAK
Wyświetlenie dostępnych rejsów	TAK	TAK	TAK
Zapis na kursy	NIE	TAK	TAK
Rezerwacja rejsów	TAK	TAK	TAK
Zarządzanie kursami i rejsami	NIE	TAK	TAK
Nauka ogólna	NIE	TAK	TAK

³ Gość – Użytkownik niezalogowany (nieautoryzowany)

Wygenerowanie egzaminu próbnego	NIE	TAK	TAK
Rozwiązywanie egzaminu próbnego	NIE	TAK	TAK
Podgląd zbliżających się kursów użytkownika	NIE	TAK	TAK
Wycofanie się z kursu	NIE	TAK	TAK
Podgląd zbliżających się rejsów użytkownika	NIE	TAK	TAK
Wycofanie się z rejsu	NIE	TAK	TAK
Wyświetlenie statystyk profilu	NIE	TAK	TAK
Edycja profilu	NIE	TAK	TAK
Wyświetlenie informacji kontaktowych	TAK	TAK	TAK
Najczęściej zadawane pytania	TAK	TAK	TAK
Panel administracyjny	NIE	NIE	TAK

3.4 Diagramy przypadków użycia

Diagram przypadków użycia (DPU), jest graficznym przedstawieniem funkcji systemu widzianych z perspektywy używających go osób. Stosuje się go, aby zilustrować interakcje i zależności pomiędzy użytkownikami (zwany dalej aktorami), a systemem, które nazywane są przypadkami użycia. Jest to narzędzie wykorzystywane głównie w analizie i projektowaniu systemów w celu określenia wymagań biznesowych. Poniższy diagram przedstawia ogólny zarys funkcjonalności systemu, które zostały dalej rozbite na konkretne przypadki użycia (PU).

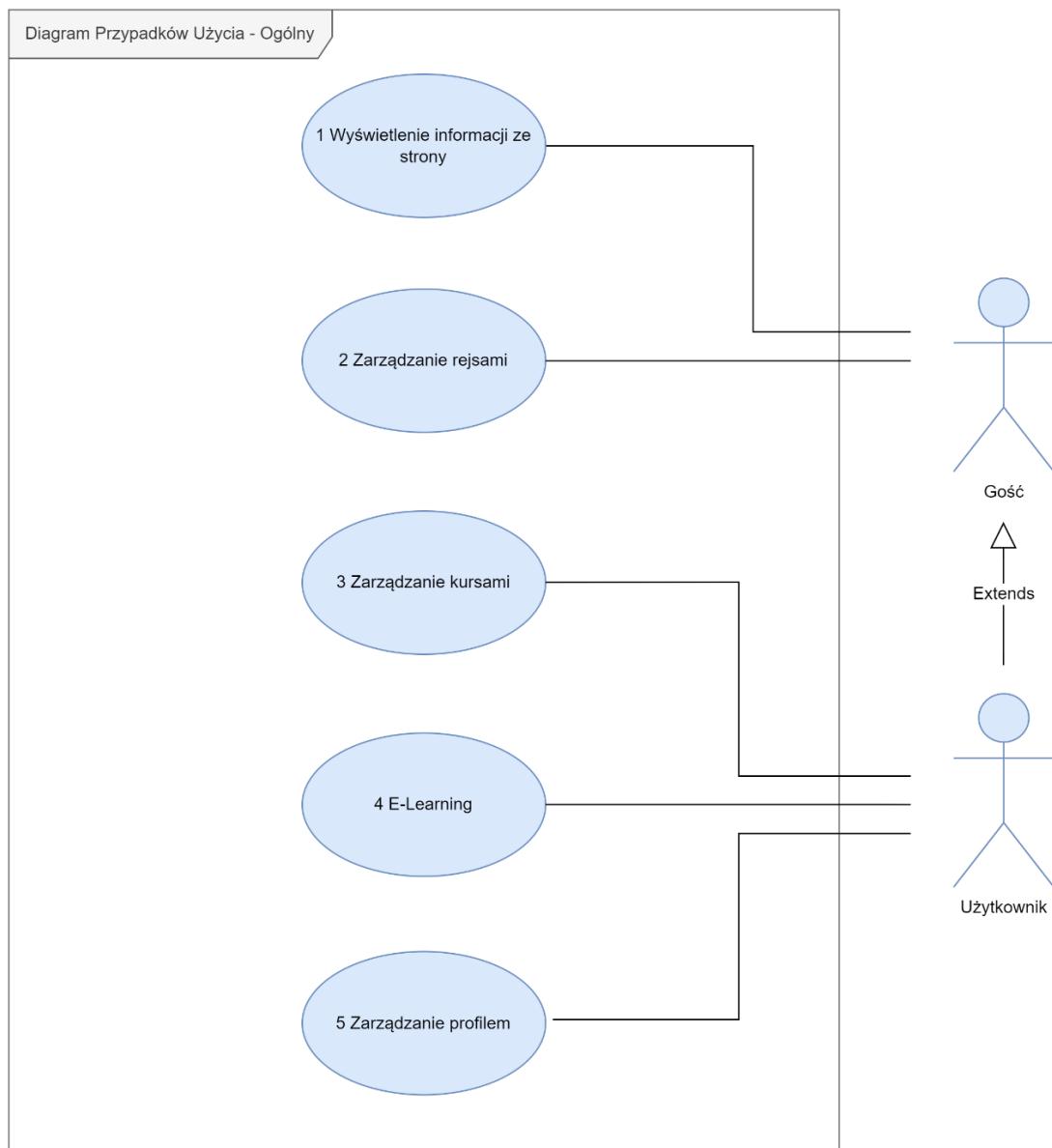


Diagram 1 - Przypadków Użycia - Ogólny

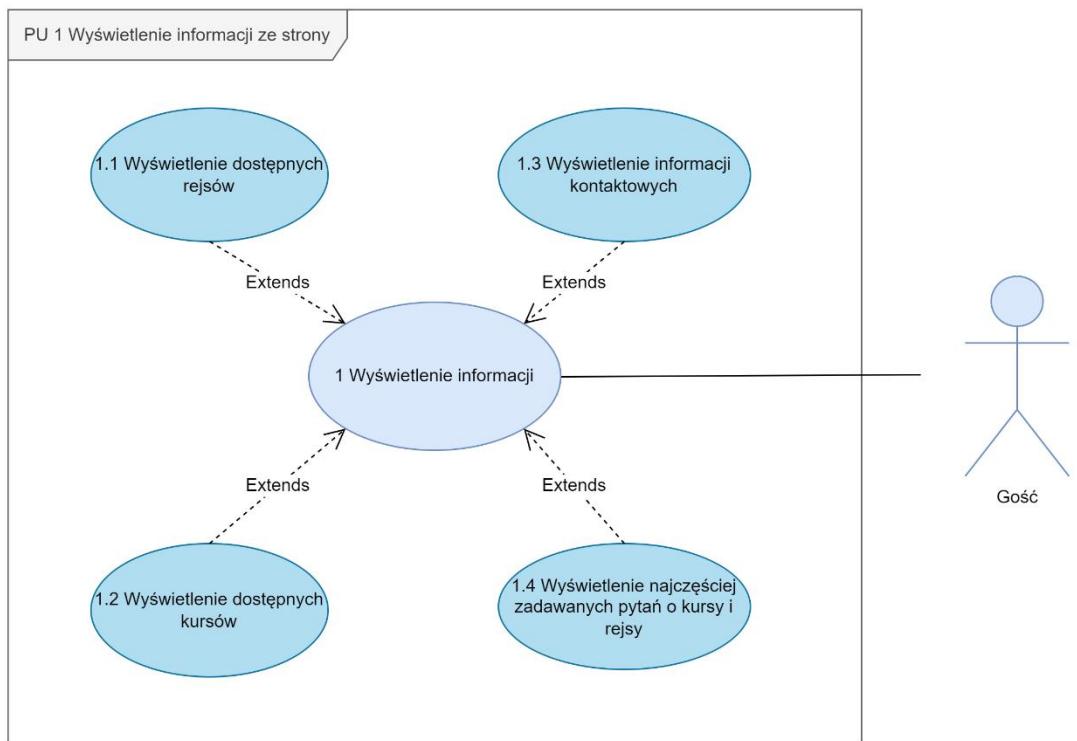


Diagram 2 - Przypadków Użycia - PU Wyświetlenie informacji ze strony

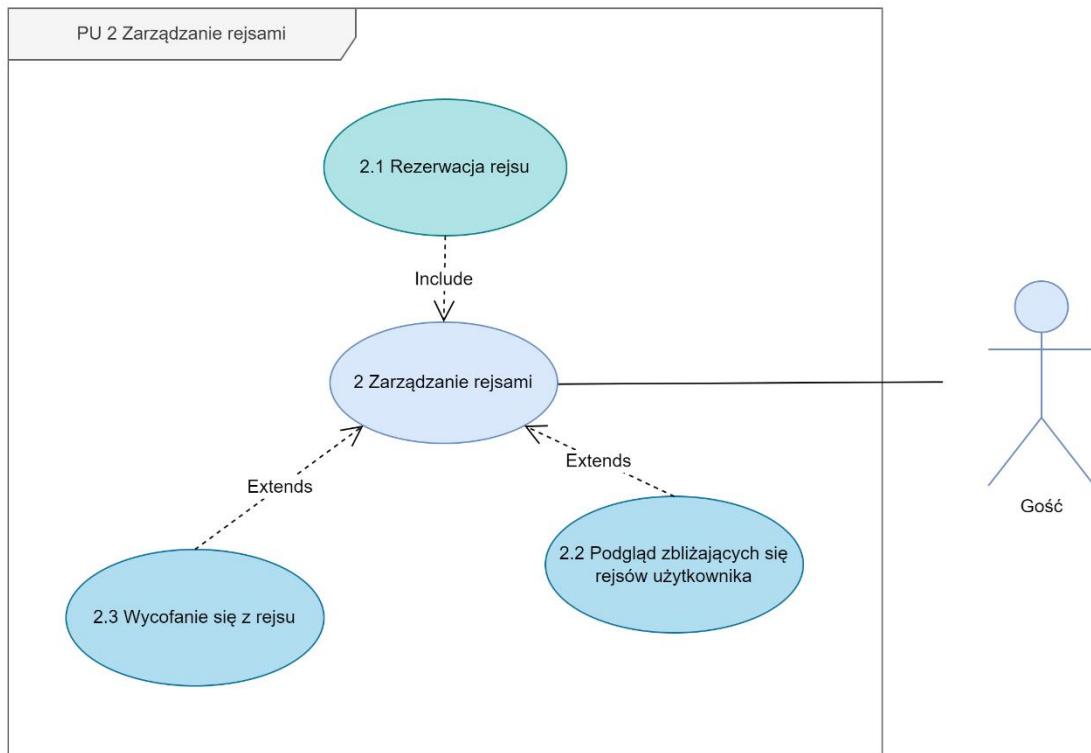


Diagram 3 - Przypadków Użycia - PU Zarządzanie rejsami

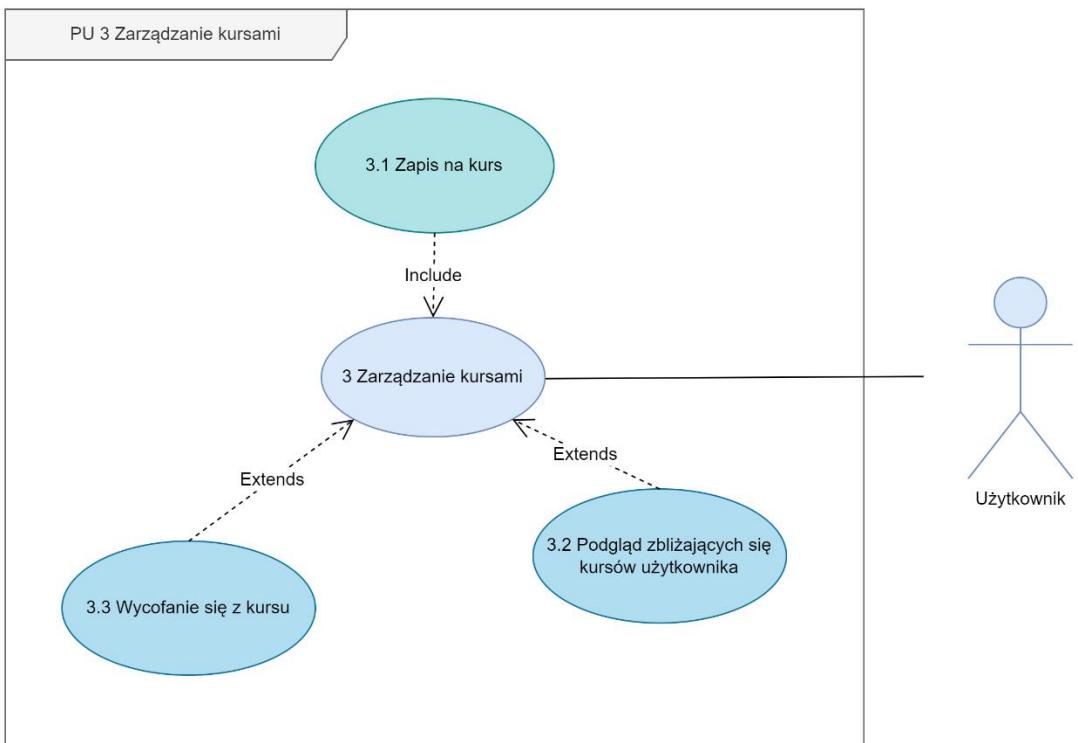


Diagram 4 - Przypadek Użycia - PU Zarządzanie kursami

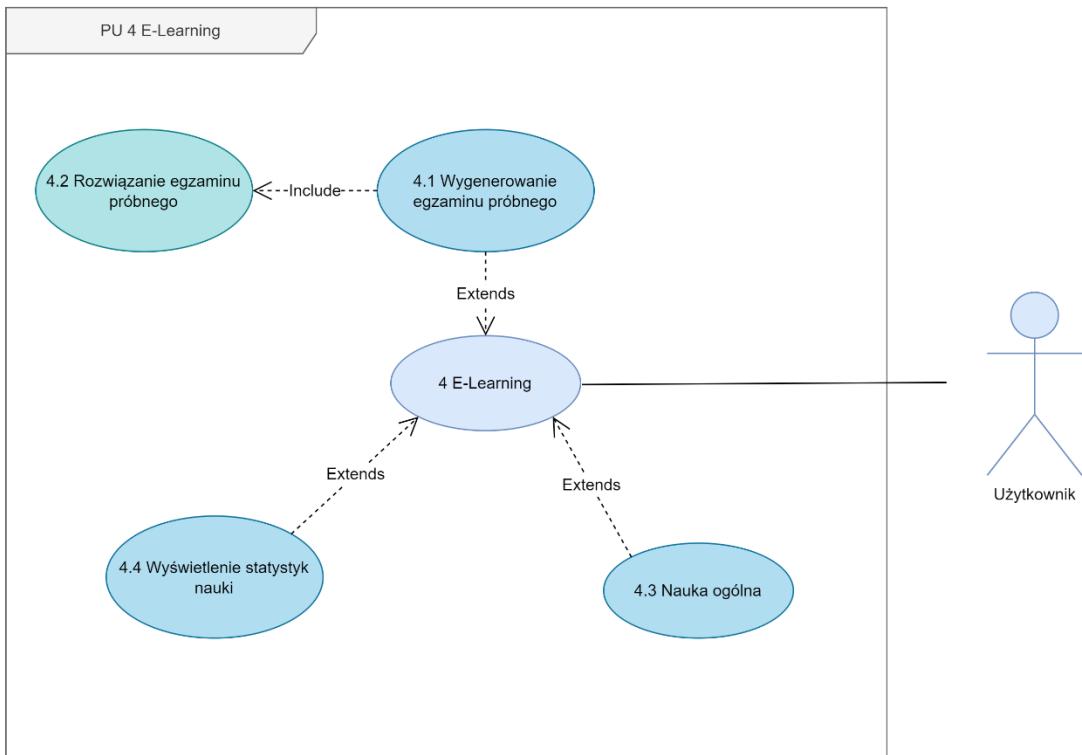


Diagram 5 - Przypadek Użycia - PU E-Learning

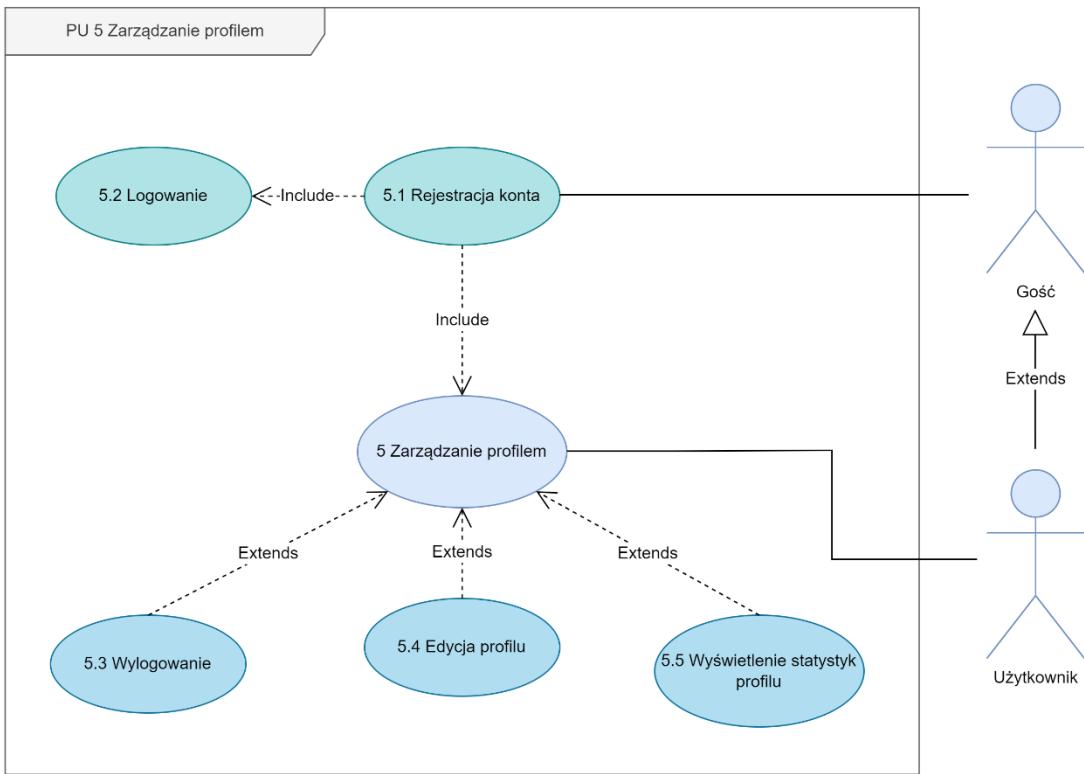


Diagram 6 - Przypadków Użycia - PU Zarządzanie profilem

3.5 Scenariusze

Scenariusze są istotnym elementem podejścia analitycznego, ze względu na:

- Projektowanie i Prototypowanie:** Służą jako podstawa do projektowania interfejsów użytkownika i prototypowania systemu. Przykładowe ścieżki użytkowników w scenariuszach mogą być bezpośrednio użyte do tworzenia prototypów i mockupów⁴.
- Definiowanie Wymagań:** Pozwalają dokładnie zdefiniować i zrozumieć wymagania użytkowników względem systemu. Przez to, że scenariusze skupią się na interakcjach użytkownika z systemem, pomagają określić, jakie funkcjonalności są potrzebne.
- Planowanie Rozwoju:** Scenariusze pomagają w planowaniu procesu rozwoju oprogramowania, umożliwiając programistom skupienie się na konkretnych funkcjach i zapewnienie, że wszystkie scenariusze użytkowania są obsługiwane.

⁴ Mockup – wizualna, dość szczegółowa reprezentacja graficznego aplikacji, która ma na celu ukazanie wyglądu i układu widoku strony

- **Testowanie:** Scenariusze stanowią podstawę dla przypadków testowych w procesie zapewniania jakości. Ewentualni testerzy mogą używać scenariuszy do weryfikacji, czy wszystkie ścieżki użytkownika są prawidłowo obsługiwane przez system. Szczególnie przydatne dla testów End to End.

Przypadek użycia: 1.1 Wyświetlenie dostępnych rejsów

Nazwa	1.1 Wyświetlenie dostępnych rejsów
Aktor	Gość
Krótki opis	Wyświetlenie konkretnego rodzaju rejsu wraz ze szczegółowymi informacjami, cennikiem i dostępnymi terminami
Warunki wstępne	Brak
Warunki końcowe	System wyświetla rejs wraz z dostępnymi terminami
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny 3a. Gość wybiera przycisk ‘Eventy’ 4. System prezentuje listę dostępnych rodzajów rejsów 5. Gość wybiera interesujący go rodzaj rejsu i przechodzi do jego szczegółów poprzez przycisk ‘Zobacz więcej’ 6. System prezentuje szczegóły wybranego rodzaju rejsu, cennik oraz dostępne terminy
Alternatywny przepływ wydarzeń	3b. Gość przechodzi na zakładkę ‘Oferta’ i wybiera ‘Eventy’

Tabela 1 - Scenariusz Przypadku Użycia 1.1 - Wyświetlenie dostępnych rejsów

Przypadek użycia: 1.2 Wyświetlenie dostępnych kursów

Nazwa	1.2 Wyświetlenie dostępnych kursów
Aktor	Gość
Krótki opis	Wyświetlenie konkretnego rodzaju kursu wraz ze szczegółowymi informacjami, cennikiem i dostępnymi terminami
Warunki wstępne	Brak
Warunki końcowe	System wyświetla kurs
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny 3a. Gość wybiera przycisk 'Kursy' 4. System prezentuje listę dostępnych rodzajów kursów 5. Gość wybiera interesujący go rodzaj kursu i przechodzi do jego szczegółów poprzez przycisk 'Zobacz więcej' 6. System prezentuje szczegóły wybranego rodzaju kursu, cennik oraz dostępne terminy
Alternatywny przepływ wydarzeń	<p>3b. Gość przechodzi na zakładkę 'Oferta' i wybiera 'Kursy'</p>

Tabela 2 - Scenariusz Przypadku Użycia 1.2 - Wyświetlenie dostępnych kursów

Przypadek użycia: 1.3 Wyświetlenie informacji kontaktowych

Nazwa	1.3 Wyświetlenie informacji kontaktowych
Aktor	Gość
Krótki opis	Wyświetlenie informacji kontaktowych potencjalnego usługodawcy
Warunki wstępne	Brak
Warunki końcowe	System wyświetla numer telefonu oraz email usługodawcy
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny

	3. Gość przechodzi na zakładkę 'O firmie' i wybiera 'Kontakt' 4. System prezentuje informacje kontaktowe
Alternatywny przepływ wydarzeń	Brak

Tabela 3 - Scenariusz Przypadku Użycia 1.3 - Wyświetlenie informacji kontaktowych

Przypadek użycia: 1.4 Wyświetlenie najczęściej zadawanych pytań o kursy i rejsy

Nazwa	1.4 Wyświetlenie najczęściej zadawanych pytań o kursy i rejsy
Aktor	Gość
Krótki opis	Wyświetlenie rozwijalnej listy wszystkich pytań i odpowiedzi na temat kursów bądź rejsów
Warunki wstępne	Brak
Warunki końcowe	System wyświetla listę pytań oraz odpowiedzi
Główny przepływ wydarzeń	1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny 3. Gość przechodzi na zakładkę 'O firmie' i wybiera 'Najczęściej zadawane pytania' 4. System prezentuje listę pytań 5. Gość wybiera interesujące go pytanie i klikna na nie 6. System prezentuje odpowiedź na wybrane pytanie
Alternatywny przepływ wydarzeń	Brak

Tabela 4 - Scenariusz Przypadku Użycia 1.4 - Wyświetlenie najczęściej zadawanych pytań o kursy i rejsy

Przypadek użycia: 2.1 Rezerwacja rejsu

Nazwa	2.1 Rezerwacja rejsu
Aktor	Gość
Krótki opis	Rezerwacja rejsu o konkretnym terminie na podane w formularzu dane osobowe

Warunki wstępne	Brak
Warunki końcowe	Zarezerwowanie konkretnego rejsu, na konkretny termin, dla konkretnego emaila, na podaną liczbę uczestników
Główny przepływ wydarzeń	<p>1. Wywołanie PU 1.1 Wyświetlenie dostępnych rejsów</p> <p>2a. Gość wybiera przycisk ‘Wybierz termin’</p> <p>3a. System prezentuje dostępne terminy</p> <p>4a. Gość wybiera interesujący go termin i kliką ‘Przejdź dalej’</p> <p>5a. System wyświetla informacje potwierdzające rezerwowany rejs, formularz do wypełnienia i końcową kwotę do zapłaty</p> <p>6a. Gość wprowadza stosowne dane do formularza oraz kliką w przycisk ‘Rezerwuj’</p> <p>7a. System powiadamia klienta o pomyślnie zarezerwowanym rejsie</p>
Alternatywny przepływ wydarzeń	<p>2b. System wyświetla szczegóły wybranego rejsu, który nie ma dostępnych terminów</p> <p>3b. Gość wraca do poprzedniego widoku w celu wybrania innego rodzaju rejsu lub porzuca proces</p> <p>4b. Gość nie wybiera żadnego terminu i anuluje proces poprzez przycisk ‘Anuluj’ bądź ‘X’</p> <p>5b. System chowa wyświetcone okno dialogowe i wraca do poprzedniego widoku</p> <p>6b. Gość wprowadza błędne dane do formularza</p> <p>7b. System przeprowadza walidację i</p>

	<p>powiadamia gości o błędnych wartościach</p> <p>6c. Gość wycofuje rezerwację rejsu poprzez przycisk ‘Anuluj’ bądź ‘X’</p> <p>7c. System chowa wyświetlony moduł i wraca do poprzedniego widoku</p>
--	--

Tabela 5 - Scenariusz Przypadku Użycia 2.1 - Rezerwacja rejsu

Przypadek użycia: 2.2 Podgląd zbliżających się rejsów użytkownika

Nazwa	2.2 Podgląd zbliżających się rejsów użytkownika
Aktor	Użytkownik
Krótki opis	Wyświetlenie listy wszystkich eventów przypisanych do emaila użytkownika
Warunki wstępne	Użytkownik jest poprawnie zalogowany
Warunki końcowe	System wyświetla listę eventów użytkownika
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny 3. Gość przechodzi na zakładkę ‘Moje eventy’ 4. System prezentuje listę eventów
Alternatywny przepływ wydarzeń	Brak

Tabela 6 - Scenariusz Przypadku Użycia 2.2 - Podgląd zbliżających się rejsów użytkownika

Przypadek użycia: 2.3 Wycofanie się z rejsu

Nazwa	2.3 Wycofanie się z rejsu
Aktor	Użytkownik
Krótki opis	Wycofanie użytkownika z wybranego rejsu
Warunki wstępne	Użytkownik jest poprawnie zalogowany i jest zarejestrowany na przynajmniej 1 rejs, który jest jeszcze nieopłacony
Warunki końcowe	System usuwa powiązanie pomiędzy użytkownikiem, a rejsem i odświeża listę rejsów

użytkownika	
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Wywołanie PU 2.2 Podgląd zbliżających się rejsów użytkownika 2. Wybiera interesujący go rejs i używa przycisku ‘Anuluj’ 3a. System powiadamia użytkownika o pomyślnym wycofaniu się z rejsu oraz odświeża listę
Alternatywny przepływ wydarzeń	<ol style="list-style-type: none"> 3b. System powiadamia użytkownika o niemożliwości wycofania się z kursu ze względu na przekroczony dopuszczalny czas do wycofania

Tabela 7 - Scenariusz Przypadku Użycia 2.3 - Wycofanie się z rejsu

Przypadek użycia: 3.1 Zapis na kurs

Nazwa	3.1 Zapis na kurs
Aktor	Użytkownik
Krótki opis	Użytkownik wpisuje się na konkretny rodzaj kursu i konkretny termin
Warunki wstępne	Użytkownik jest poprawnie zalogowany
Warunki końcowe	Wpisanie użytkownika na dany kurs
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Wywołanie PU 1.2 Wyświetlenie dostępnych kursów 2a. Użytkownik przechodzi do procesu zapisu na kurs poprzez przycisk ‘Zapisz się na kurs’ 3. System prezentuje dostępne terminy 4a. Użytkownik wybiera interesujący go termin i kliką ‘Zapisz się’ 5a. System wyświetla informacje potwierdzające zapis na kurs
Alternatywny przepływ wydarzeń	<ol style="list-style-type: none"> 2b. System wyświetla szczegóły wybranego kursu, który nie ma dostępnych terminów 3b. Użytkownik wraca do poprzedniego widoku w celu wybrania innego rodzaju

	<p>kursu lub porzuca proces</p> <p>4b. Użytkownik nie wybiera żadnego terminu i anuluje proces poprzez przycisk ‘Anuluj’ bądź ‘X’</p> <p>5b. System chowa wyświetcone okno dialogowe i wraca do poprzedniego widoku</p> <p>4c. Użytkownik wybiera termin, na który już się zapisał i kliką ‘Zapisz się’</p> <p>5c. System wyświetla stosowny komunikat i odrzuca zapis na kurs</p>
--	--

Tabela 8 - Scenariusz Przypadku Użycia 3.1 - Zapis na kurs

Przypadek użycia: 3.2 Podgląd zbliżających się kursów użytkownika

Nazwa	3.2 Podgląd zbliżających się kursów użytkownika
Aktor	Użytkownik
Krótki opis	Wyświetlenie listy wszystkich kursów przypisanych do konta użytkownika
Warunki wstępne	Użytkownik jest poprawnie zalogowany
Warunki końcowe	System wyświetla listę kursów użytkownika
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny 3. Gość przechodzi na zakładkę ‘Moje kursy’ 4. System prezentuje listę kursów
Alternatywny przepływ wydarzeń	Brak

Tabela 9 - Scenariusz Przypadku Użycia 3.2 - Podgląd zbliżających się kursów użytkownika

Przypadek użycia: 3.3 Wycofanie się z kursu

Nazwa	3.3 Wycofanie się z kursu
Aktor	Użytkownik

Krótki opis	Wypisanie użytkownika z wybranego kursu
Warunki wstępne	Użytkownik jest poprawnie zalogowany i jest zapisany na przynajmniej 1 kurs, który jest jeszcze nieopłacony
Warunki końcowe	System usuwa powiązanie pomiędzy użytkownikiem, a kursem i odświeża listę kursów użytkownika
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Wywołanie PU 3.2 Podgląd zbliżających się kursów użytkownika 2. Wybiera interesujący go kurs i używa przycisku ‘Anuluj’ 3a. System powiadamia użytkownika o pomyślnym wycofaniu się z rejsu oraz odświeża listę
Alternatywny przepływ wydarzeń	<ol style="list-style-type: none"> 3b. System powiadamia użytkownika o niemożliwości wycofania się z kursu ze względu na jego opłacenie

Tabela 10 - Scenariusz Przypadku Użycia 3.3 - Wycofanie się z kursu

Przypadek użycia: 4.1 Wygenerowanie egzaminu próbnego

Nazwa	4.1 Wygenerowanie egzaminu próbnego
Aktor	Użytkownik
Krótki opis	Wygenerowanie użytkownikowi wszystkich pytań do egzaminu oraz rozpoczęcie odliczania czasu
Warunki wstępne	Użytkownik jest poprawnie zalogowany i jest zapisany na przynajmniej 1 kurs, który jest opłacony
Warunki końcowe	System generuje egzamin próbny
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1a. Wywołanie PU 3.2 Podgląd zbliżających się kursów użytkownika 2. Użytkownik wybiera interesujący go

	<p>kurs i używa przycisku ‘Nauka’</p> <ol style="list-style-type: none"> 3. System wyświetla listę kursów i dostępnych przycisków do e-learningu⁵ 4. Użytkownik używa przycisku do wygenerowania próbnego egzaminu dla interesującego go opłaconego kursu 5. System generuje egzamin próbny oraz rozpoczyna odliczanie do jego zakończenia
Alternatywny przepływ wydarzeń	1b. Użytkownik przechodzi na zakładkę ‘Nauka’

Tabela 11 - Scenariusz Przypadku Użycia 4.1 - Wygenerowanie egzaminu próbnego

Przypadek użycia: 4.2 Rozwiązywanie egzaminu próbnego

Nazwa	4.2 Rozwiązywanie egzaminu próbnego
Aktor	Użytkownik
Krótki opis	Użytkownik odpowiada na pytania oraz kończy egzamin próbny
Warunki wstępne	Użytkownik jest poprawnie zalogowany i jest zapisany na przynajmniej 1 kurs, który jest opłacony
Warunki końcowe	System wyświetla wyniki z rozwiązanego egzaminu oraz dopisuje go do statystyk
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Wywołanie PU 4.1 Wygenerowanie egzaminu próbnego 2. Użytkownik kończy wybrany egzamin poprzez przycisk ‘Zakończ egzamin’ 3. System wyświetla wyniki z rozwiązanego egzaminu
Alternatywny przepływ wydarzeń	Brak

Tabela 12 - Scenariusz Przypadku Użycia 4.2 - Rozwiązywanie egzaminu próbnego

⁵ Przyciski do e-learningu to statystyki nauki, nauka ogólna (wygenerowanie losowego pytania) oraz egzamin próbny prowadzący do jego wygenerowania

Przypadek użycia: 4.3 Nauka ogólna

Nazwa	4.3 Nauka ogólna
Aktor	Użytkownik
Krótki opis	Użytkownik dostaje losowe pytanie do rozwiązyania z konkretnego działu żeglarskiego
Warunki wstępne	Użytkownik jest poprawnie zalogowany i jest zapisany na przynajmniej 1 kurs, który jest opłacony
Warunki końcowe	System generuje losowe pytanie
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1a. Wywołanie PU 3.2 Podgląd zbliżających się kursów użytkownika 2. Użytkownik wybiera interesujący go kurs i używa przycisku ‘Nauka’ 3. System wyświetla listę kursów i dostępnych przycisków do e-learningu 4. Użytkownik używa przycisku do nauki ogólnej dla interesującego go opłaconego kursu 5. System generuje losowe pytanie 6a. Użytkownik odpowiada na pytanie i sprawdza odpowiedź 7a. System wyświetla poprawną odpowiedź
Alternatywny przepływ wydarzeń	<ol style="list-style-type: none"> 1b. Użytkownik przechodzi na zakładkę ‘Nauka’ 6b. Użytkownik przechodzi do następnego pytania 7b. System generuje kolejne pytanie

Tabela 13 - Scenariusz Przypadku Użycia 4.3 - Nauka ogólna

Przypadek użycia: 4.4 Statystyki nauki

Nazwa	4.4 Statystyki nauki
Aktor	Użytkownik
Krótki opis	Użytkownik otrzymuje spersonalizowane statystyki nauki

Warunki wstępne	Użytkownik jest poprawnie zalogowany i jest zapisany na przynajmniej 1 kurs opłacony
Warunki końcowe	System generuje wyniki z e-learningu
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1a. Wywołanie PU 3.2 Podgląd zbliżających się kursów użytkownika 2. Użytkownik wybiera interesujący go kurs i używa przycisku ‘Nauka’ 3. System wyświetla listę kursów i dostępnych przycisków do e-learningu 4. Użytkownik używa przycisku do statystyki nauki dla interesującego go opłaconego kursu 5. System generuje statystyki
Alternatywny przepływ wydarzeń	Brak

Tabela 14 - Scenariusz Przypadku Użycia 4.4 - Statystyki nauki

Przypadek użycia: 5.1 Rejestracja konta

Nazwa	5.1 Rejestracja konta
Aktor	Gość
Krótki opis	Utworzenie konta poprzez formularz danych osobowych
Warunki wstępne	Brak
Warunki końcowe	System tworzy nowego użytkownika
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny 3. Gość przechodzi na zakładkę ‘Zarejestruj’ z poziomu górnego paska nawigacyjnego 4. System prezentuje panel rejestracyjny 5a. Gość wprowadza poprawne dane i kliką przycisk ‘Zarejestruj się’ 6a. System powiadamia użytkownika o pomyślnej rejestracji i przechodzi do panelu logowania

Alternatywny przepływ wydarzeń	<p>5b. Gość wprowadza niepoprawne dane</p> <p>6b. System powiadamia o wprowadzeniu niepoprawnych danych w konkretnym polu</p> <p>5c. Gość wycofuje się na stronę główną dzięki przyciskowi ‘Powrót do strony głównej’</p> <p>6c. System wycofuje proces rejestracji i przekierowuje na stronę główną aplikacji</p>
--------------------------------	--

Tabela 15 - Scenariusz Przypadku Użycia 5.1 - Rejestracja konta

Przypadek użycia: 5.2 Logowanie

Nazwa	5.2 Logowanie
Aktor	Gość
Krótki opis	Autoryzacja oraz uwierzytelnienie gościa oraz nadanie mu statusu użytkownika
Warunki wstępne	Brak
Warunki końcowe	System loguje gościa w aplikacji i nadaje mu uprawnienia do poszczególnych funkcjonalności
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Gość przechodzi na stronę główną aplikacji 2. System prezentuje panel powitalny 3. Gość przechodzi na zakładkę ‘Zaloguj się’ z poziomu górnego paska nawigacyjnego 4. System prezentuje panel logowania z formularzem 5a. Gość wprowadza poprawne dane i kliką przycisk ‘Zaloguj się’ 6a. System powiadamia użytkownika o pomyślnie przeprowadzonym logowaniu i przechodzi na stronę główną aplikacji

Alternatywny przepływ wydarzeń	<p>5b. Gość wprowadza niepoprawne dane</p> <p>6b. System powiadamia o wprowadzeniu niepoprawnych danych w konkretnym polu</p> <p>5c. Gość wycofuje się na stronę główną dzięki przyciskowi ‘Powrót do strony głównej’</p> <p>6c. System wycofuje proces logowania i przekierowuje na stronę główną aplikacji</p>
--------------------------------	--

Tabela 16 - Scenariusz Przypadku Użycia 5.2 - Logowanie

Przypadek użycia: 5.3 Wylogowanie

Nazwa	5.3 Wylogowanie
Aktor	Użytkownik
Krótki opis	Usunięcie autoryzacji oraz uwierzytelnienia dla aktualnie zalogowanego użytkownika oraz przywrócenie mu statusu gościa
Warunki wstępne	Użytkownik jest aktualnie zalogowany
Warunki końcowe	System wylogowuje użytkownika w aplikacji i odbiera mu uprawnienia do poszczególnych funkcjonalności
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Użytkownik z poziomu górnego paska nawigacyjnego kliką w przycisk ‘Wyloguj się’ 2. System powiadamia użytkownika o pomyślnym wylogowaniu 3. System przekierowuje użytkownika na stronę główną aplikacji
Alternatywny przepływ wydarzeń	Brak

Tabela 17 - Scenariusz Przypadku Użycia 5.3 - Wylogowanie

Przypadek użycia: 5.4 Edycja profilu

Nazwa	5.4 Edycja profilu
Aktor	Użytkownik
Krótki opis	Edytowanie danych osobowych profilu użytkownika
Warunki wstępne	Użytkownik jest aktualnie zalogowany
Warunki końcowe	System zmienia podane w formularzu dane osobowe
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Użytkownik z poziomu górnego paska nawigacyjnego kliką w przycisk 'Mój profil' 2. System wyświetla formularz do edycji danych osobowych 3a. Użytkownik podaje dane osobowe o poprawnym formacie 4a. Użytkownik podaje poprawne hasło w celach weryfikacyjnych 5a. System powiadamia użytkownika o pomyślnej zmianie danych
Alternatywny przepływ wydarzeń	<ol style="list-style-type: none"> 3b. Użytkownik podaje dane osobowe o niepoprawnym formacie 4b. System powiadamia użytkownika o źle wypełnionych danych 4c. Użytkownik podaje niepoprawne hasło 5b. System powiadamia użytkownika o niepoprawnym haśle

Tabela 18 - Scenariusz Przypadku Użycia 5.4 - Edycja profilu

Przypadek użycia: 5.5 Wyświetlenie statystyk profilu

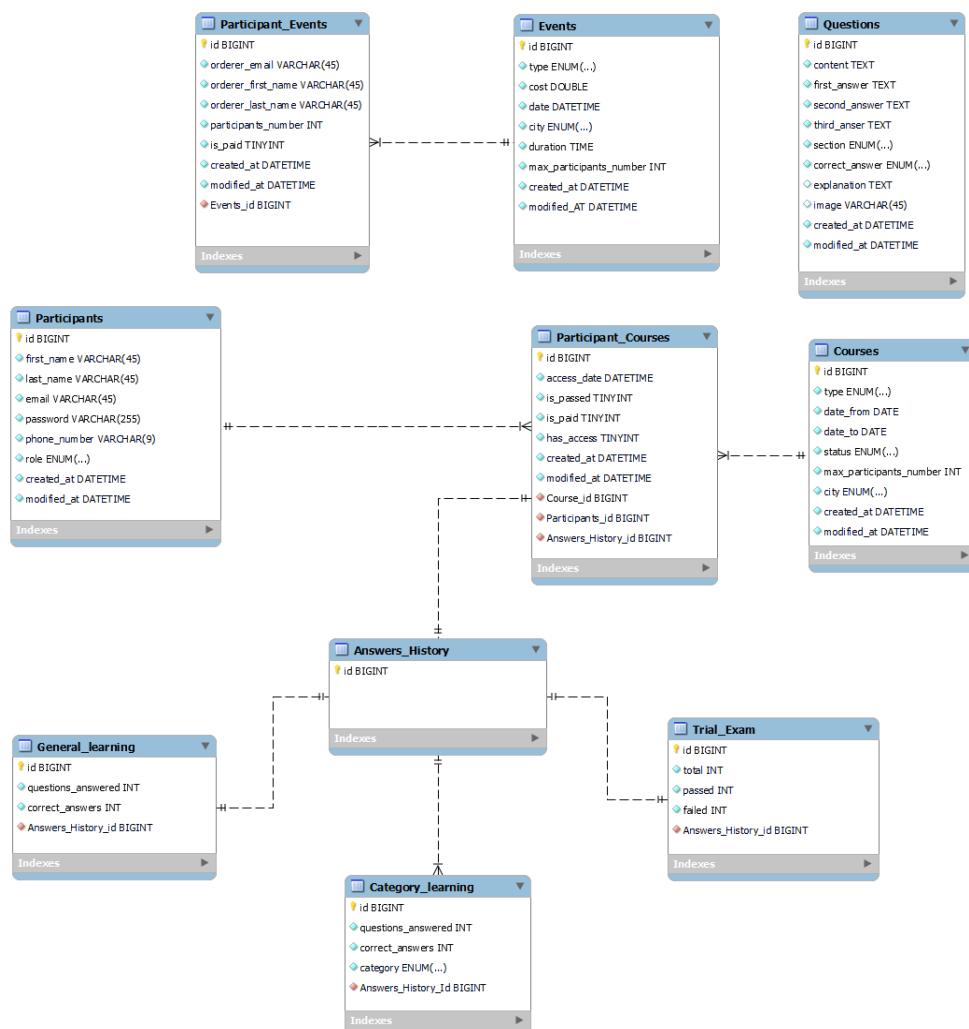
Nazwa	5.5 Wyświetlenie statystyk profilu
Aktor	Użytkownik
Krótki opis	Wyświetlenie statystyk profilu aktualnie zalogowanego użytkownika
Warunki wstępne	Użytkownik jest aktualnie zalogowany
Warunki końcowe	System wyświetla statystyki profilu
Główny przepływ wydarzeń	<ol style="list-style-type: none"> 1. Użytkownik z poziomu górnego paska nawigacyjnego kliką w przycisk ‘Mój profil’ 2. System wyświetla panel statystyk profilu
Alternatywny przepływ wydarzeń	Brak

Tabela 19 - Scenariusz Przypadku Użycia 5.5 - Wyświetlenie statystyk profilu

4. Projekt bazy danych

4.1 Schemat bazy danych

Wizualizacja bazy danych (najlepiej w formie pełnoprawnego schematu), jest kluczowym etapem w procesie tworzenia każdej aplikacji internetowej, stanowiąc fundament, na którym opiera się struktura całego systemu. W dobie internetu, gdzie dane stanowią prawdopodobnie najważniejszą walutę, przemyślane zarządzanie nimi jest istnie niezbędne. Dzięki efektywnej, skalowalnej architekturze, system może być rozwijany w przyszłości bez konieczności przebudowy całej bazy danych, co oszczędza szczególnie czas i zasoby. Schematy baz danych umożliwiają nie tylko wizualizację struktury danych, ale również służą jako ważne narzędzie komunikacyjne wśród zespołów deweloperskich i tak też było w tym przypadku, gdzie schemat bazy danych musiał być zaplanowany od strony administratora, i klienta.



Rysunek 1 - Schemat bazy danych

4.2 Opis szczegółowy encji i relacji bazy danych

W każdej encji powtarzają się trzy pola, są to odpowiednio:

- **id** – jest to klucz główny encji typu BIGINT, który pełni rolę identyfikatora każdego rekordu
- **created_at** – pole typu DATETIME informujące o szczegółowym terminie utworzenia danego rekordu, automatycznie przydzielane przy dodaniu nowego rekordu do tabeli, w której omawiane pole jest zadeklarowane
- **modified_at** – pole typu DATETIME informujące o szczegółowym terminie modyfikacji danego rekordu, automatycznie przydzielana przy edycji rekordu w tabeli, w której omawiane pole jest zadeklarowane

a także typy danych:

- **BIGINT** - Typ numeryczny używany do przechowywania dużych liczb całkowitych. W systemach baz danych, BIGINT może przechowywać liczby od -2^{63} do $2^{63}-1$
- **INT** – Typ całkowitoliczbowy, który jest stosowany do przechowywania liczb całkowitych o standardowym rozmiarze. Może przechowywać wartości od -2^{31} do $2^{31}-1$
- **DOUBLE** - Typ zmiennoprzecinkowy, który służy do przechowywania liczb z dużą precyzją po przecinku
- **VARCHAR** – Typ danych tekstowych o możliwości zadeklarowania maksymalnej długości, co bywa wygodne, gdy dane tekstowe, które chcemy przechowywać, mogą mieć różną długość, ponieważ zajmuje tylko tyle miejsca, ile jest potrzebne do przechowywania rzeczywistych danych
- **TEXT** – Typ danych służący do przechowywania dużych ilości tekstu. W przeciwnieństwie do VARCHAR, TEXT umożliwia przechowywanie większej ilości tekstu, co czasami bywa obowiązkowe, ale kosztem tego zajmuje stałą ilość miejsca w bazie danych, co może być mniej wydajne
- **TINYINT** – Reprezentuje bardzo mały zakres liczb całkowitych, od -128 do 127. W przypadku omawianego projektu wykorzystywany do przechowywania wartości logicznych (prawda lub fałsz), gdzie 0 oznacza wartości fałszu a 1 oznacza prawdę

- **DATE** – Służy do przechowywania daty w formacie YYYY-MM-DD bez informacji o dokładnym czasie. Wykorzystywany, gdy interesuje nas tylko konkretny dzień, bez godzin czy minut
- **TIME** – Używany do przechowywania informacji o czasie, bez daty w formacie HH:MM:SS
- **DATETIME** – Typ danych przeznaczony do przechowywania zarówno daty, jak i czasu, zazwyczaj w formacie YYYY-MM-DD HH:MM:SS
- **ENUM** - Typ wyliczeniowy, który pozwala przechowywać wcześniej zdefiniowane ciągi tekstowe w formie ENUM ('przykład1', 'przykład2', 'przykład3')

- Encja Questions

Pole	Typ danych	Co przechowuje
content	TEXT	Treść pytania
first_answer	TEXT	Zawartość odpowiedzi A
second_answer	TEXT	Zawartość odpowiedzi B
third_answer	TEXT	Zawartość odpowiedzi C
section	ENUM	Kategorię pytania
correct_answer	ENUM	Poprawną odpowiedź
explanation	TEXT	Wyjaśnienie odpowiedzi
image	VARCHAR(45)	Ścieżkę do zdjęcia

Tabela 20 - Opis pól i relacji encji Questions

- Encja Events

Pole	Typ danych	Co przechowuje
type	ENUM	Rodzaj przeprowadzanego rejsu
cost	DOUBLE	Koszt rejsu na 1 osobę
date	DATETIME	Datę wypłynięcia
city	ENUM	Lokalizację wypłynięcia
duration	TIME	Długość trwania rejsu
max_participants_number	INT	Maksymalna liczba uczestników na dany rejs

Tabela 21 - Opis pól i relacji encji Events

- Encja Participant events

Pole	Typ danych	Co przechowuje
orderer_email	VARCHAR(45)	Adres poczty elektronicznej klienta rezerwującego rejs
orderer_first_name	VARCHAR(45)	Imię klienta rezerwującego rejs
orderer_last_name	VARCHAR(45)	Nazwisko klienta rezerwującego rejs
participants_number	INT	Liczba osób dla których zarezerwowany został rejs
is_paid	TINYINT	Wartość logiczną, czy zarezerwowany rejs został już opłacony
events_id	BIGINT	Klucz obcy ⁶ , który relacjonuje tę tabelę z tabelą events w celu zdefiniowania jaki rejs został zarezerwowany przez konkretnego klienta

Tabela 22 - Opis pól i relacji encji Participant events

- Encja Participants

Pole	Typ danych	Co przechowuje
first_name	VARCHAR(45)	Imię zarejestrowanego użytkownika aplikacji
last_name	VARCHAR(45)	Nazwisko zarejestrowanego użytkownika aplikacji
email	VARCHAR(45)	Adres poczty elektronicznej zarejestrowanego użytkownika
password	VARCHAR(255)	Hasło zarejestrowanego użytkownika aplikacji. Przechowuje maksymalnie 255 znaków ze względu na sposób zabezpieczenia hasła na bazie danych.

⁶ Klucz obcy – Pole, które jest kluczem głównym w innej tabeli. Służy do połączenia w relacji jedną tabelę z drugą.

phone_number	VARCHAR(9)	Numer telefonu użytkownika
role	ENUM	Rola w aplikacji użytkownika, dzieli się na klienta i administratora systemu

Tabela 23 - Opis pól i relacji encji Participants

- Encja Participant courses

Pole	Typ danych	Co przechowuje
access_date	DATETIME	Datę otrzymania dostępu do e-learningu
is_passed	TINYINT	Wartość logiczną, czy konkretny kurs użytkownika został zaliczony
is_paid	TINYINT	Wartość logiczną, czy konkretny kurs użytkownika został opłacony
has_access	TINYINT	Wartość logiczną, czy konkretny kurs użytkownika jest udostępniony do e-learningu
course_id	BIGINT	Klucz obcy, który relacjonuje tę tabelę z tabelą courses w celu zdefiniowania, który kurs przynależy do konkretnego użytkownika
participants_id	BIGINT	Klucz obcy, który relacjonuje tę tabelę z tabelą participants w celu zdefiniowania, który użytkownik przynależy do konkretnego kursu
answers_history_id	BIGINT	Klucz obcy, który relacjonuje tę tabelę z tabelą answers history w celu zdefiniowania konkretnych statystyk nauki do kursu użytkownika

Tabela 24 - Opis pól i relacji encji Participant courses

- Encja Courses

Pole	Typ danych	Co przechowuje
type	ENUM	Rodzaj przeprowadzanego kursu
date_from	DATE	Datę rozpoczęcia kursu
date_to	DATE	Datę zakończenia kursu
status	ENUM	Status przeprowadzanego kursu
max_participants_number	INT	Maksymalną liczbę uczestników dla jakiej można przeprowadzić kurs podczas jednego terminu
city	ENUM	Miejsce przeprowadzanego kursu

Tabela 25 - Opis pól i relacji encji Courses

- Encja Answers history jest połączona z kursami użytkownika oraz konkretnym rodzajem nauki (nauka ogólna, nauka kategoriami lub egzamin próbny), w celu stworzenia unikalnego rekordu śledzącego historię odpowiedzi użytkownika w ramach opłaconego kursu, co pozwoli następnie na szeroką analizę statystyki nauki dla interesanta.
- Encja General learning

Pole	Typ danych	Co przechowuje
questions_answered	INT	Liczbę wszystkich udzielonych odpowiedzi
correct_answers	INT	Liczbę poprawnie udzielonych odpowiedzi
answers_history_id	BIGINT	Klucz obcy, który relacjonuje tę tabelę z tabelą answers_history w celu zdefiniowania historii odpowiedzi z nauki ogólnej dla całej statystyki elearningu

Tabela 26 - Opis pól i relacji encji General learning

- Encja Category learning

Pole	Typ danych	Co przechowuje
questions_answered	INT	Liczbę wszystkich udzielonych odpowiedzi
correct_answers	INT	Liczbę poprawnie udzielonych odpowiedzi
category	ENUM	Kategorię udzielonych odpowiedzi
answers_history_id	BIGINT	Klucz obcy, który relacjonuje tę tabelę z tabelą answers_history w celu zdefiniowania historii odpowiedzi z nauki kategoriami dla całej statystyki elearningu

Tabela 27 - Opis pól i relacji encji Category learning

- Encja Trial exam

Pole	Typ danych	Co przechowuje
total	INT	Liczbę wszystkich rozpoczętych egzaminów próbnych
passed	INT	Liczbę zaliczonych egzaminów próbnych
failed	INT	Liczbę niezaliczonych egzaminów próbnych
answers_history_id	BIGINT	Klucz obcy, który relacjonuje tę tabelę z tabelą answers_history do zdefiniowania historii odpowiedzi z egzaminów dla całej statystyki elearningu

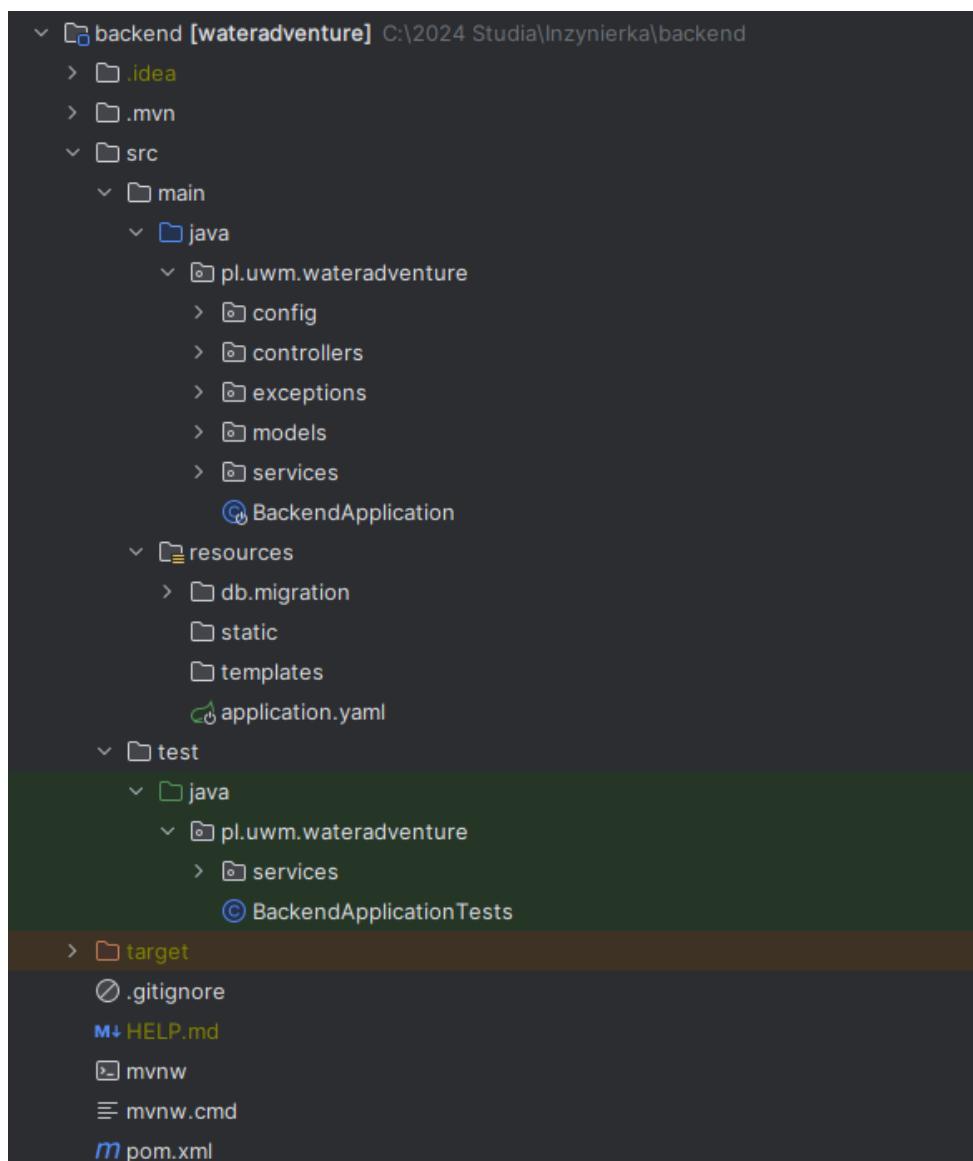
Tabela 28 - Opis pól i relacji encji Trial exam

5. Implementacja

5.1 Architektura backendu

W rozdziale dotyczącym sposobu implementacji backendu w niniejszej aplikacji webowej, skupię się na zastosowaniu frameworka Spring Boot omijając charakterystykę pisania kodu w języku Java.

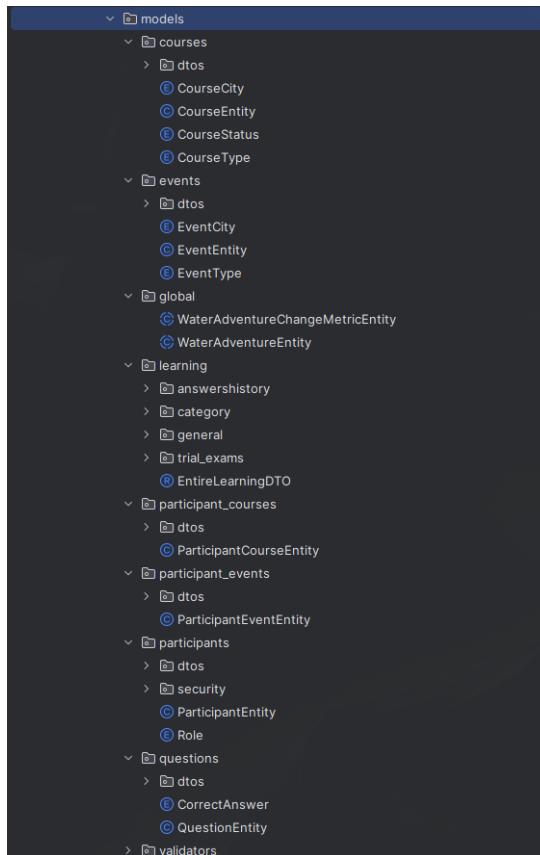
Backend aplikacji jest sercem systemu, obsługując logikę biznesową, operacje na danych oraz interakcje z bazą danych. Wykorzystanie Spring Boot pozwala na efektywne zbudowanie skalowalnej i bezpiecznej aplikacji serwerowej, która komunikuje się z frontendem za pomocą REST API. Jego struktura wygląda następująco:



Rysunek 2 - Ogólna struktura backendu aplikacji

Można w niej dostrzec, że dzieli się na główne pakiety, które zawierają w sobie poszczególne rodzaje encji, wygląda to następująco:

1. **Models** – Reprezentuje strukturę danych aplikacji, zawiera klasy odzwierciedlające tabele bazodanowe, z którymi operuje aplikacja. Modele te korzystają z anotacji JPA do mapowania obiektowo-relacyjnego, co umożliwia łatwą komunikację z bazą danych. Jak widać na rysunku numer 3, dla modeli stworzone zostały oddzielne pakiety dla poszczególnych encji, których zawartość dzieli się na klasy typu:
 - **Entity** - Klasa przechowująca model encji odzwierciedlający tabelę bazodanową, jej atrybuty i ewentualne metody.
 - **DTO** - Klasy tego typu służą do przesyłania danych między backendem a frontendem. Są używane do uniknięcia nadmiernego przesyłania niepotrzebnych informacji i pozwalają na kontrolowane udostępnianie tylko tych danych, które są potrzebne w kontekście danej funkcjonalności.
 - **Enum (oznaczone jako symbol ‘E’ na rysunku numer 1)** - Używane do reprezentowania stałych zbiorów wartości, które mają konkretne przeznaczenie. Są przydatne, gdy istnieje ograniczony zestaw opcji lub kategorii. W niniejszej aplikacji użyte dla między innymi określenia typu przeprowadzanych kursów.



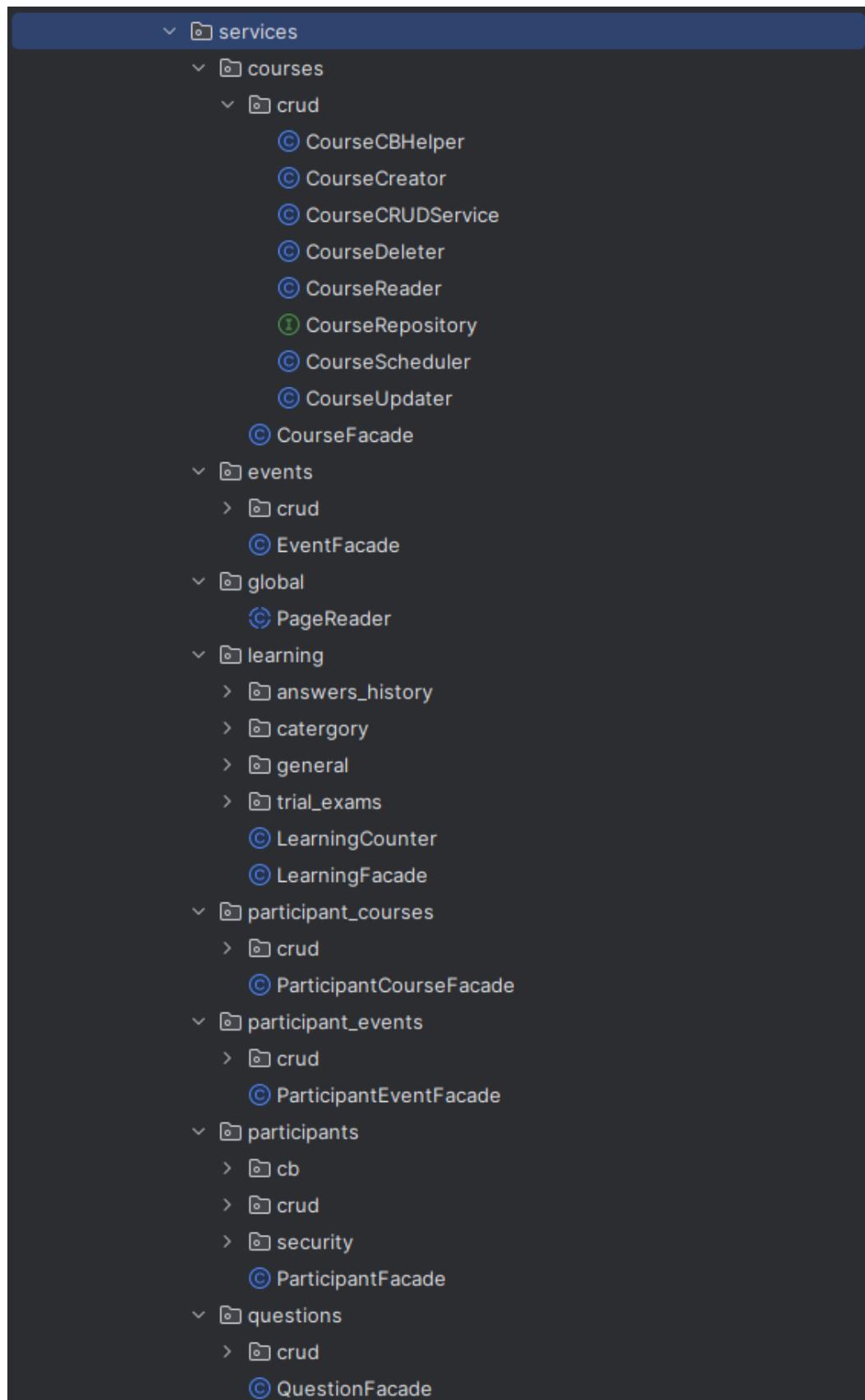
Rysunek 3 - Struktura pakietu models

2. **Services** – Zawiera logikę systemu aplikacji, przetwarzając dane i zarządzając przepływem informacji między repozytoriami, a kontrolerami. W serwisach wykorzystywane jest wstrzykiwanie zależności (Dependency Injection), które jest kluczową cechą Springa, pozwalającą na łatwe zarządzanie wszelkimi komponentami aplikacji. Pakiet services dzieli się na klasy niższego poziomu, takie jak :

- **Creator** – Klasy dostarczające logikę niezbędną do tworzenia rekordów w bazie danych wraz z ewentualną validacją.
- **Reader** – Klasy umożliwiające przetwarzanie i odczyt danych z encji poprzez metody do pobierania pojedynczych obiektów lub list obiektów na podstawie podanych kryteriów.
- **Updater** – Klasy odpowiedzialne za aktualizację istniejących rekordów i ich danych wraz z ewentualną validacją.
- **Deleter** – Klasy zawierające metody do usuwania pojedynczych, wskazanych rekordów.
- **Repository** – Repozytorium modelu encji, są to interfejsy zapewniające abstrakcyjną warstwę danych, które wykorzystują JPA (Java Persistence API) do automatyzacji operacji bazodanowych.
- **CBHelper** – Umożliwia efektywne filtrowanie rekordów z bazy danych.

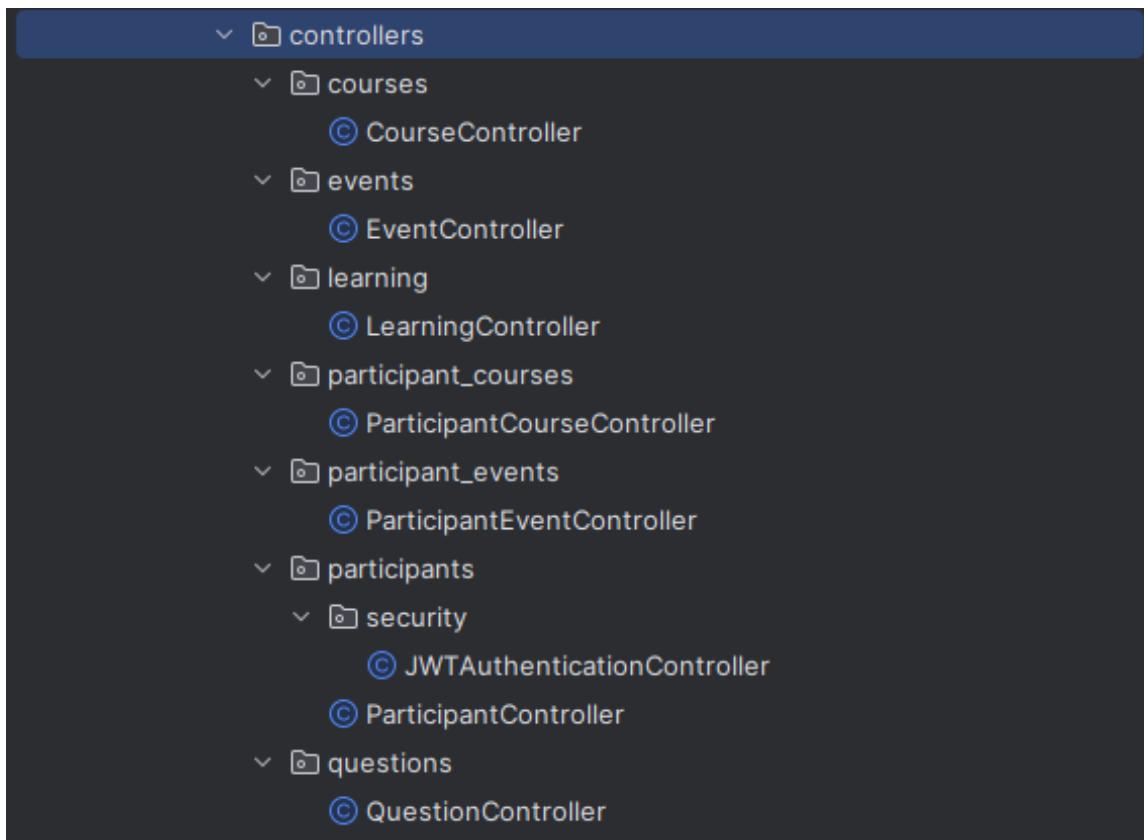
oraz klasy wyższego poziomu:

- **CRUDService** – Łączy cały CRUD w jedność, co umożliwia wywoływanie ich z innych poziomów aplikacji (domyślnie klasy Creator, Reader, Updater i Deleter są możliwe do użycia tylko na poziomie pakietu, w którym się znajdują).
- **Facade** – Służy do tworzenia spójnego interfejsu wyższego poziomu, który może być używany przez inne części aplikacji, takie jak kontrolery. Facade wywołuje odpowiednie metody z CRUService i serwisów innych modeli. Ułatwia to kontrolę nad operacjami i zapewnia spójność w obszarze obsługi różnych modeli.
- **Security** – Sekcja zawierająca klasy obsługujące logikę Spring Security. Odpowiedzialna za zabezpieczenie autoryzacji i uwierzytelnienia użytkownika w aplikacji, a także generowanie dla niego odpowiedniego tokenu w sesji.



■ Rysunek 4 - Struktura pakietu services

3. **Controllers** - Pakiet zawierający wszelkie kontrolery dla modeli. Kontrolery odpowiadają za obsługę żądań HTTP, komunikując się z klientem przez REST API⁷. Definiują punkty końcowe (endpoints), do których frontend może wysyłać żądania, a następnie zwracać odpowiedzi. Stanowią zatem pomost między logiką backendu, a frontendem.

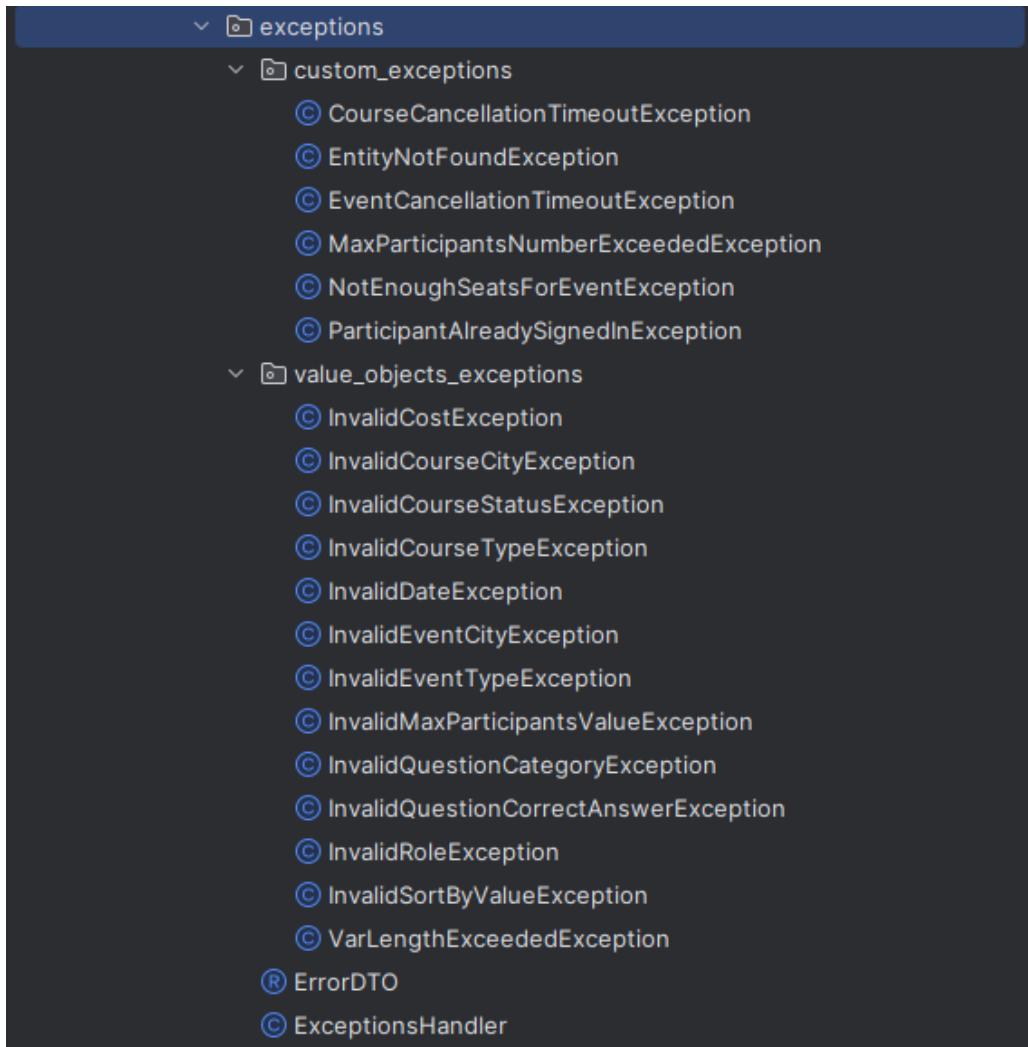


Rysunek 5- Struktura pakietu controllers

⁷**REST API (Representational State Transfer)** - Architektura stosowana do projektowania sieciowych interfejsów programowania aplikacji (API), które pozwalają na komunikację pomiędzy aplikacjami w sposób zrozumiały i skalowalny. REST stanowi innowacyjne podejście do tworzenia API w nowoczesnych aplikacjach webowych.

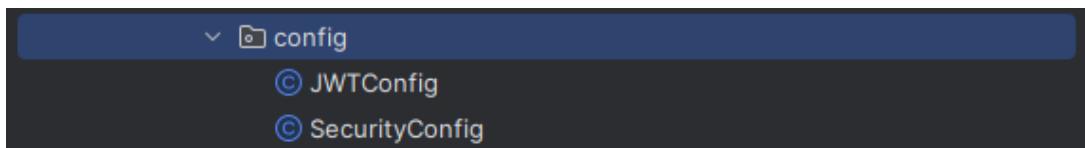
4. **Exceptions** – Zestaw utworzony w celu przechowywania możliwie występujących wyjątków (błędów) aplikacyjnych oraz ich kontrolowaną obsługę przy wysyłaniu żądań HTTP. Zawiera przede wszystkim klasy typu:

- **Exception** – Klasy definiujące konkretny wyjątek, jego wiadomość oraz ewentualne pole, które doprowadziło do wyrzucenia wyjątku.
- **ExceptionsHandler** – Centralny komponent do obsługi wyjątków, który pozwala na kontrolowaną reakcję i wysyłanie spójnych odpowiedzi do klienta.



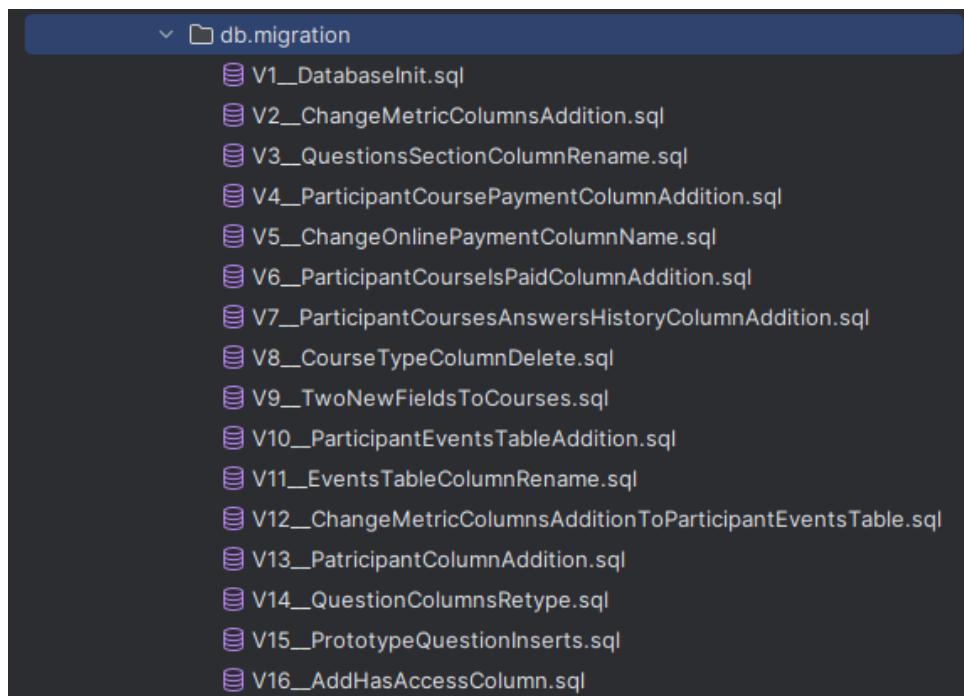
Rysunek 6 - Struktura pakietu exceptions

5. **Config** – Pakiet zawierający konfigurację dotyczącą JWT (JWT Config) oraz Spring Security (SecurityConfig). Te konfiguracje są niezbędne do zapewnienia bezpieczeństwa aplikacji i ochrony jej zasobów. JWTConfig zajmuje się konfiguracją uwierzytelniania na podstawie JWT, podczas gdy SecurityConfig ustala zasady dostępu do poszczególnych części aplikacji oraz obsługuje zarządzanie sesjami i inne aspekty bezpieczeństwa. Dzięki tym konfiguracjom aplikacja jest chroniona przed nieautoryzowanym dostępem i zagrożeniami bezpieczeństwa.



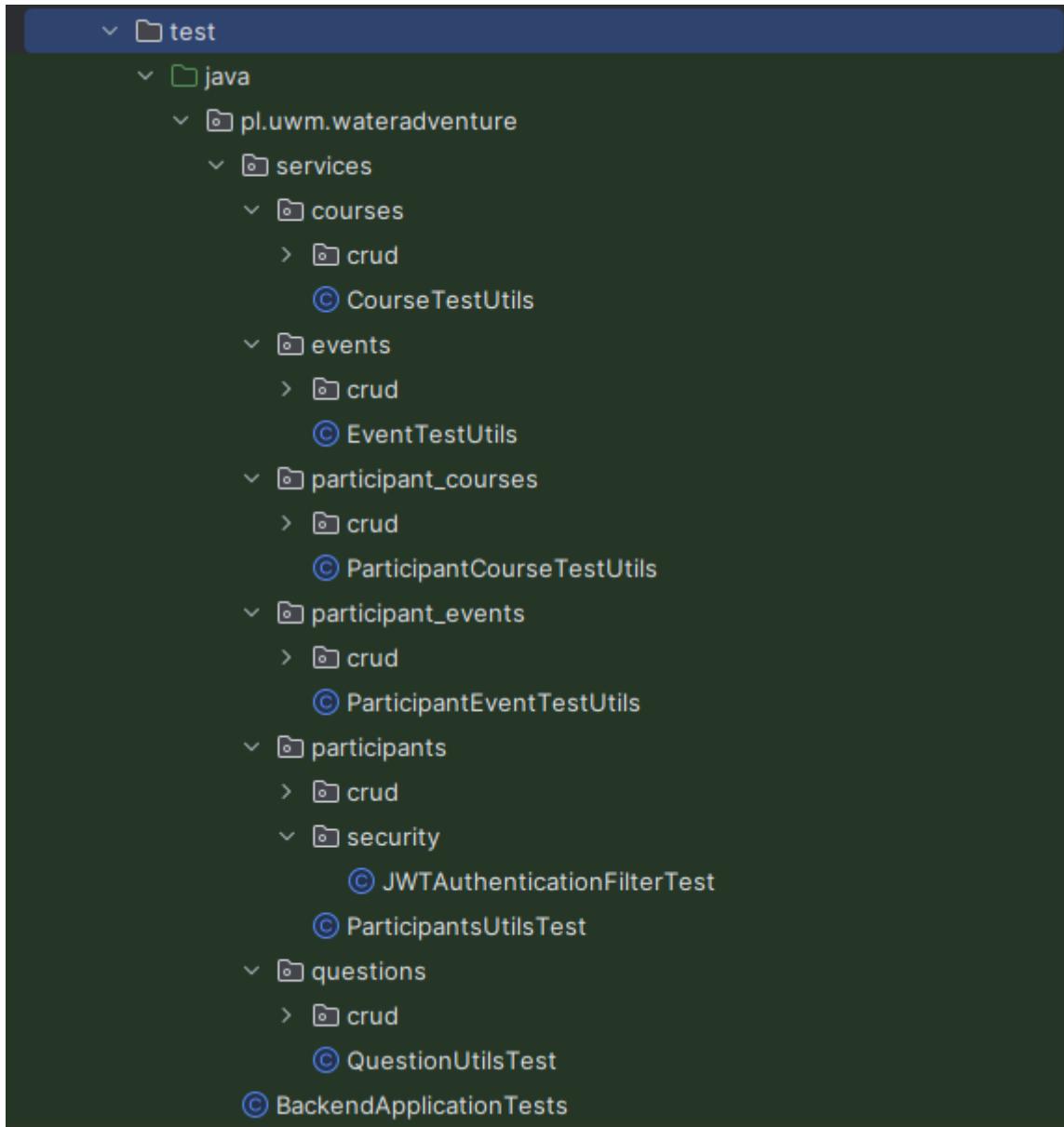
Rysunek 7- Struktura pakietu config

6. **Migration** – Przechowuje pliki migracyjne, czyli skrypty SQL, wdrażane bezpośrednio na bazę danych. Definiują zmiany w schemacie bazy danych, takie jak tworzenie tabel, dodawanie kolumn czy modyfikacja istniejących struktur. Dzięki narzędziu Flyway, każda zmiana schematu jest wersjonowana i stosowana w kontrolowany sposób, co zapewnia spójność bazy danych w różnych środowiskach i łatwość śledzenia zmian.



Rysunek 8 - Struktura pakietu migration

7. **Test** - Dedykowane miejsce, gdzie umieszczane są testy jednostkowe. Testy jednostkowe to skrypty automatycznie uruchamiane przy wdrażaniu aplikacji, wykonujące sprawdzenie pojedynczych części kodu, zwykle metod lub klas, w izolacji od reszty systemu. Celem testów jednostkowych jest zapewnienie, że każda jednostka kodu działa poprawnie i zgodnie z oczekiwaniemi. Testy omówione zostaną szczegółowo w późniejszym rozdziale pracy dyplomowej.



Rysunek 9 - Struktura pakietu test

5.2 Architektura frontendu

Typowa architektura aplikacji napisanej przy użyciu frameworka Angular opiera się na wzorcu architektonicznym nazwanym Model-View-Controller (MVC), który jest często określany jako Model-View-ViewModel (MVVM) w kontekście aplikacji webowych. Angular oferuje własną implementację tego wzorca, ale z pewnymi specyficznymi nazwami dla elementów architektury. Oto omówienie typowej architektury Angulara:

1. Komponenty (Components):

W projekcie Angularowym, komponenty są centralnym elementem architektury. Każda strona lub widok aplikacji jest reprezentowana przez komponent. Każdy komponent składa się z trzech głównych części: klasy Typescript, szablonu HTML i pliku stylów CSS (bądź innych arkuszy stylów). Komponenty są odpowiedzialne za logikę biznesową oraz interakcję z użytkownikiem. Hierarchiczne zagnieźdzanie komponentów pozwala na budowanie skomplikowanych interfejsów z mniejszych, ponownie wykorzystywanych elementów, co przyczynia się do modułowości i reużywalności kodu.

2. Serwisy (Services):

Serwisy to klasy, które służą do dzielenia się logiką biznesową niezwiązaną bezpośrednio z interfejsem użytkownika między różnymi komponentami. Serwisy są używane do wykonywania operacji, takich jak pobieranie danych z serwera, zarządzanie stanem aplikacji, obsługa autoryzacji, itp. Serwisy są wstrzykiwane podobnie jak po stronie backendowej napisanej w Spring Boot (dependency injection) do komponentów i są tworzone jako singletony, co oznacza, że istnieje tylko jedna instancja serwisu w całej aplikacji.

3. Moduły Routingowe:

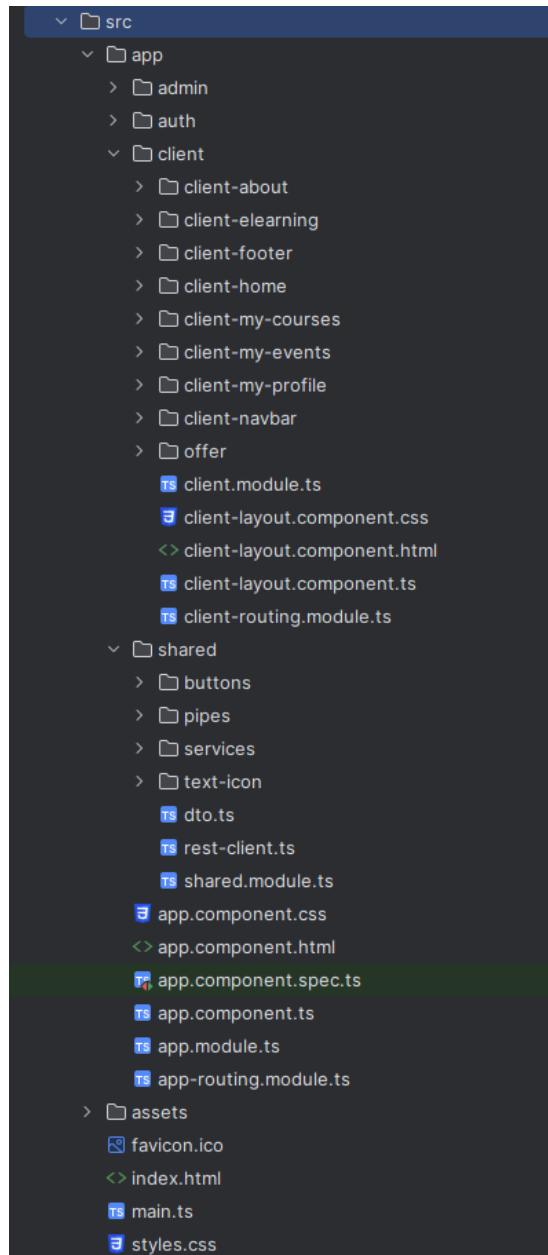
Moduł routingu pozwala na zarządzanie nawigacją między różnymi widokami w aplikacji. Umożliwia tworzenie trasy (Route) dla każdej strony lub widoku. Zmieniając trasę w przeglądarce, można łatwo nawigować między widokami bez przeładowywania strony. Moduły routingowe pozwalają ponadto na zorganizowanie reguł nawigacji, obsługę parametrów URL oraz zabezpieczenie dostępu do określonych części aplikacji.

4. Moduły (Modules):

Moduły służą do organizacji komponentów, dyrektyw, serwisów i innych części aplikacji w logiczne grupy. Każda aplikacja Angular ma główny moduł (AppModule), który służy do uruchomienia aplikacji. Dodatkowe moduły mogą być tworzone w celu lepszej organizacji

kodu i funkcji, co ułatwia zarządzanie aplikacją, szczególnie gdy jej rozmiar rośnie. W analizowanym projekcie, moduły zostały podzielone zgodnie z podziałem głównych funkcjonalności aplikacji, co pozwoliło na większą intuicyjność i łatwość zarządzania kodem. Moduły takie jak auth dla autoryzacji, admin dla panelu administracyjnego, client dla panelu klienta i shared dla współdzielonych zasobów i usług, zapewniają logiczną organizację kodu, co jest szczególnie istotne w skalowalnych i rozbudowanych aplikacjach.

Omawiana architektura ma oczywiście odzwierciedlenie w omawianej aplikacji, co wygląda następująco:



Ilustracja 10 - Struktura ogólna frontendu aplikacji

Struktura pakietu jest modularna, co oznacza, że poszczególne funkcjonalności są podzielone na mniejsze, łatwe do zarządzania części, takie jak:

- **Auth** – Zawiera komponenty odpowiedzialne za proces logowania i rejestracji. To tutaj znajduje się odwołanie do logiki bezpieczeństwa z backendu. Pakiet ten przechowuje także komponent do obsługi błędów 404, który pozwala na eleganckie zarządzanie sytuacjami, gdy użytkownik dostaje się na stronę w domenie aplikacji, która nie istnieje.
- **Client** – Centrum dla wszystkich komponentów panelu klienta. W tym miejscu umieszczane są wszelkie widoki i elementy interfejsu, które umożliwiają użytkownikowi interakcję z aplikacją (np. przeglądanie dostępnych kursów i rejsów).
- **Shared** - Zawiera elementy wspólne dla różnych części aplikacji, takie jak serwisy i komponenty wielokrotnego użytku. W tym pakiecie umieszczane są narzędzia, które nie są specyficzne dla jednego modułu, lecz mogą być wykorzystywane przez zarówno panel administratora i klienta. Obejmuje logikę do obsługi ciasteczek (cookies), przekierowań (routing) i zarządzania stanem aplikacji. Zawiera najistotniejsze w kwestii backendu klasy, czyli tę przechowującą wszelkie DTO (dto.ts), które są odzwierciedleniem tych z backendu oraz klasę metod wysyłających żądania na konkretny adres HTTP, które wywołują funkcje z backendu (rest-client.ts).

5.3 Projektowanie logotypu

Logotyp, często postrzegany jako wizytówka marki, jest kluczowym elementem każdej aplikacji webowej. Jego znaczenie wykracza daleko poza samą estetykę – dobrze zaprojektowany logotyp ma fundamentalne znaczenie dla komunikacji wizualnej, identyfikacji marki oraz jej pozycjonowania na rynku.

Znaczenie Logotypu:

1. **Pierwsze Wrażenie:** W świecie aplikacji webowych, gdzie użytkownik może szybko przełączać się między stronami, logo to często pierwszy element, na który zwraca uwagę. Stwarza ono pierwsze, często nieświadome, wrażenie marki. Dobre logo ma potencjał przyciągnąć uwagę i zachęcić do bliższego poznania aplikacji.
2. **Tożsamość:** Logo jest wizualnym streszczeniem tożsamości marki. Powinno ono komunikować użytkownikom, kim jesteś jako firma, co oferujesz i jakie wartości

reprezentujesz. Spójne logo, które jest zgodne z misją firmy i jej wartościami, wzmacnia tożsamość marki.

3. **Unikalność:** W przesyconym rynku aplikacji webowych, gdzie nowe produkty pojawiają się niemal każdego dnia, unikatowe i rozpoznawalne logo pomaga wyróżnić Twoją aplikację od konkurencji. Jest to kluczowy element budowania silnej marki, która pozostaje w pamięci użytkowników.

Projekt graficzny:



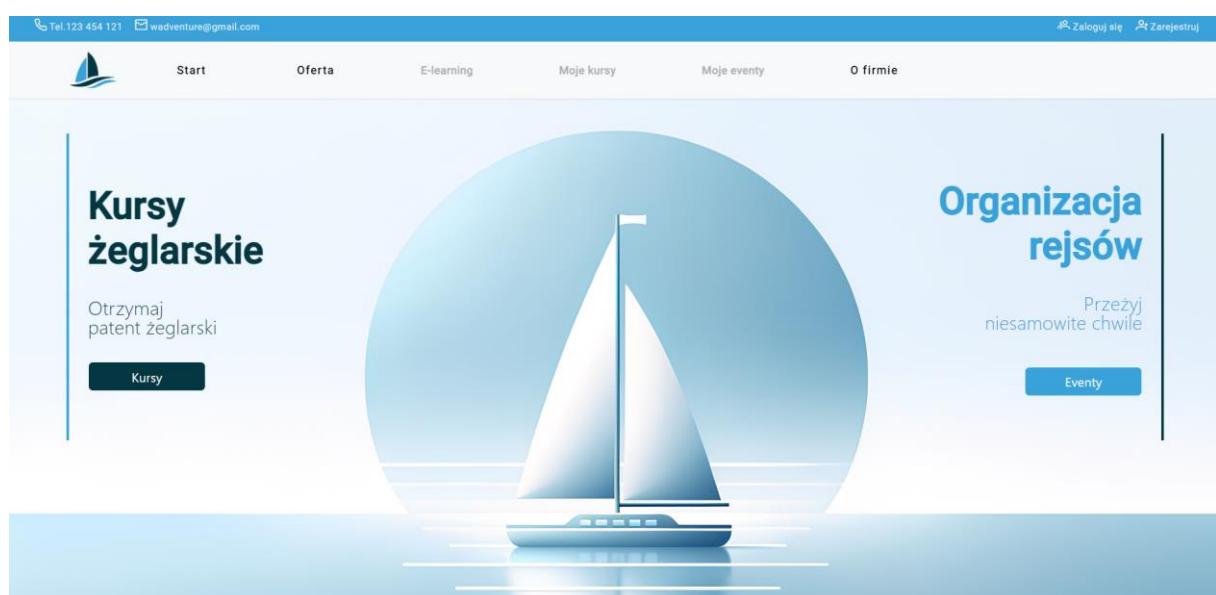
Rysunek 11 - Logotyp aplikacji

Logo zostało wykreowane w programie Adobe Photoshop 2021 i dostosowane do charakteru omawianej aplikacji internetowej, aby jak najbardziej kojarzyło się z szeroko pojętym żeglowaniem.

6. Instrukcja użytkownika aplikacji

W dzisiejszym cyfrowym świecie, intuicyjność i łatwość obsługi aplikacji webowej stają się nie tylko atutem, ale także koniecznością. Dobrze zaprojektowany interfejs użytkownika oraz przemyślana ścieżka użytkownika są kluczowe dla zapewnienia pozytywnych doświadczeń. Niniejsza instrukcja ma na celu dokładne zaprezentowanie tego, jak użytkownicy mogą nawigować i korzystać z funkcji aplikacji, kierując ich przez każdy element interfejsu krok po kroku.

6.1 Strona główna



Rysunek 12 - Instrukcja użytkownika - Strona główna

Na panelu powitalnym, widocznym na rysunku numer 12, możemy wyróżnić od samej góry dwuczęściowy pasek nawigacyjny, który zawiera:

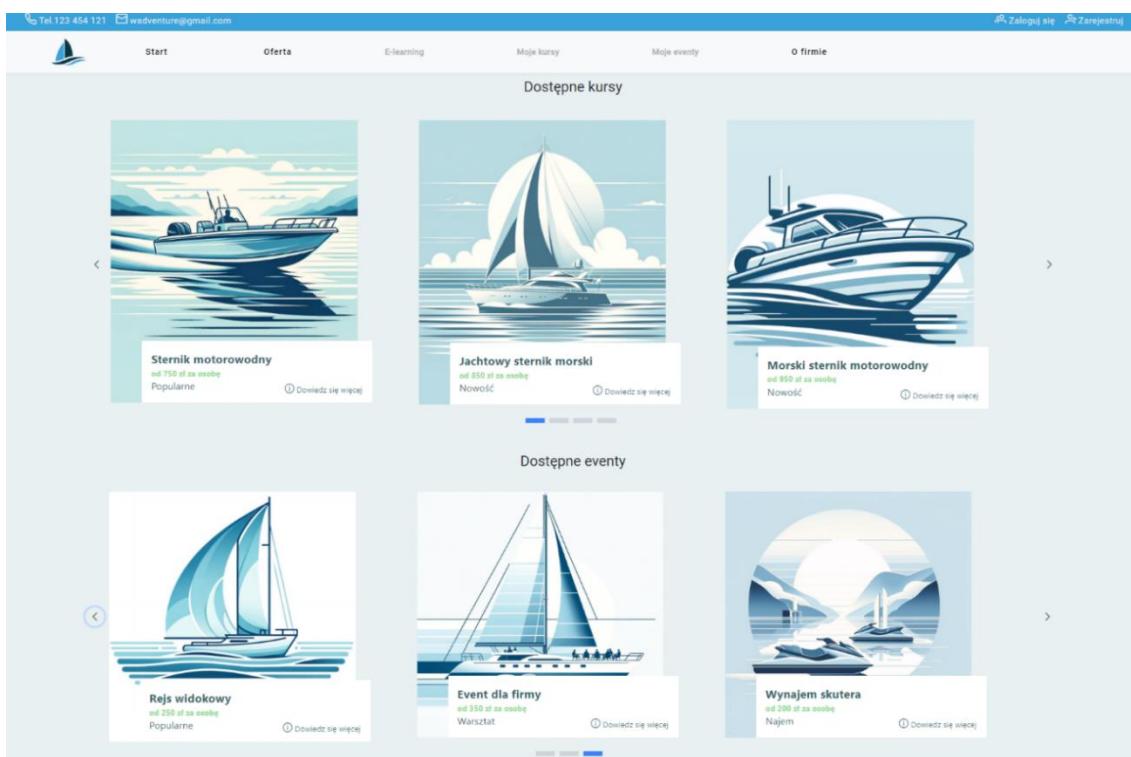


Rysunek 13 - Instrukcja użytkownika - Pasek nawigacyjny

- **Numer telefonu oraz e-mail** – klikane pola, które umożliwiają bezpośrednie wykonanie połączenia komórkowego lub wysłania wiadomości poczty elektronicznej
- **Przycisk Zaloguj się** – przekierowuje na stronę autoryzacji i uwierzytelnienia, gdzie możemy się zalogować przy pomocy formularza danych

- **Przycisk Zarejestruj** – przekierowuje na widok tworzenia nowego użytkownika
- **Logotyp aplikacji**
- **Przycisk Start** – ekspresowy powrót na stronę główną
- **Zakładka Oferta** – pozwala na przejście do listy kursów lub listy rejsów
- **Zakładka E-learning (nieaktywna)** – przekierowanie do modułu nauki zdalnej, z której możemy skorzystać dla opłaconych kursów
- **Zakładka Moje kursy (nieaktywna)** – widok wszystkich kursów użytkownika
- **Zakładka Moje eventy (nieaktywna)** – widok wszystkich rejsów przypisanych do emailu zalogowanego użytkownika
- **Zakładka o firmie** – pozwala na przejście do informacji kontaktowych lub listy FAQ

Przechodząc niżej, jako że strona główna jest dosyć sporych rozmiarów, ukazują nam się dwie automatyczne karuzele, które pozwalają na przejrzenie oferty w szybszy sposób niż przechodzenie na zakładkę z ofertą, ale i mniej szczegółowy:



Rysunek 14- Instrukcja użytkownika - Strona główna - Część druga

Idąc dalej mamy informacje zachęcające użytkownika do skorzystania z oferty:

The screenshot shows the homepage of the Dlaczego Water Adventure website. At the top, there's a header with a phone number (Tel. 123 454 121), an email address (wadventure@gmail.com), and links for 'Zaloguj się' (Log in) and 'Zarejestruj' (Register). Below the header is a navigation menu with links to 'Start', 'Oferta', 'E-learning', 'Moje kursy', 'Moje eventy', and 'O firmie'. A logo of a sailboat is on the left. The main content area features a title 'Dlaczego Water Adventure?' above two columns of bullet points. The left column lists: 'Szeroki zakres usług od warsztatów po kursy na patent sternika', 'Zdobywasz umiejętności z doświadczoną kadrą', 'Indywidualne podejście do klienta', and 'Dzięki nam zdobywasz praktyczną wiedzę'. The right column lists: 'Gwarancja relaksu i dobrych zabawy', 'Możliwość podziwiania pięknych widoków', 'Catering i dobra muzyka na pokładzie', and 'Rejsy zimą i latem'. Below this is a testimonial box with a photo of a person in a boat, the name 'Hubert Wróblewski-Klimaszewski', and a quote: 'Zdawalność egzaminów na wysokim poziomie, satysfakcja z każdego przebytego rejsu - to dzięki temu uwielbiam wykonywać swoją pracę. Jeśli chcesz zostać samodzielnym sternikiem bądź przesyć niesamowitą przygodę poza lądem, to serdecznie zachęcam do skorzystania z moich usług.' A 'Więcej o mnie' button is at the bottom.

Rysunek 15- Instrukcja użytkownika - Strona główna - Część trzecia

Znajdziemy także informacje o lokalizacji przeprowadzanej żeglugi i opinie klientów, które podobnie jak oferta są automatycznie przewijane w formie karuzeli:

The screenshot shows a testimonial section with a heading 'Skąd wypływamy' (From where we sail). It displays two locations: 'Sopot' and 'Olecko'. Each location has a photo, address, and contact information. Below this is a section titled 'Sprawdź opinię naszych usług' (Check our service reviews) showing three reviews from Mikolaj Graczyk, Marcin Frej, and Andrzej Szewczak. Each review includes a star rating, date, and a short comment. At the bottom, there's a note about reviews coming from Facebook and a link to go back.

Rysunek 16 - Instrukcja użytkownika - Strona główna - część czwarta

Na samym dole znajduje się stopka aplikacji, która przede wszystkim pełni funkcję informacyjną i podobnie jak pasek nawigacyjny widoczny na rysunku numer 13, jest przytwierdzona do każdego widoku strony internetowej, aby użytkownik mógł łatwo i wygodnie dostać się do obu komponentów:



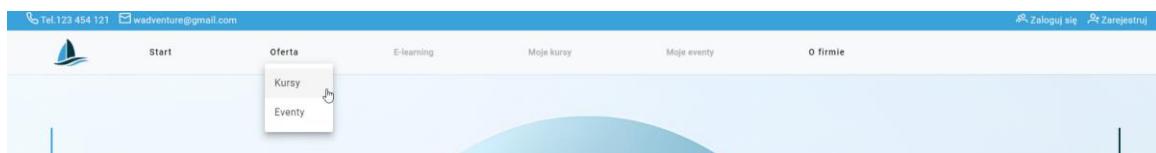
Rysunek 17 - Instrukcja użytkownika - Stopka

Na stopce znajdują się:

- Przycisk do powrotu na góre strony (panel powitalny)
- Dane kontaktowe
- Linki do social mediów
- Logotyp aplikacji w innej kolorystyce i większym rozmiarze
- Moduły Google Maps dla dwóch lokalizacji przeprowadzanej żeglugi
- Informacje dopełniające, takie jak treści o prawie autorskim oraz motto aplikacji

6.2 Zakładka Oferta

Wracając do paska nawigacyjnego z rysunku numer 13, który jest swoistym menu całej aplikacji, mamy zakładkę z ofertą żeglarską, która pozwala na przejście do listy dostępnych kursów i eventów:



Rysunek 18- Instrukcja użytkownika - Pasek nawigacyjny - Zakładka Oferta

Po przejściu zaś na listę kursów, otrzymujemy widok:

Rysunek 19- Instrukcja użytkownika - Lista kursów

Który pozwala na zapoznanie się z ogólną ofertą kursów oraz przejście do ich szczegółów:

Rysunek 20 - Instrukcja użytkownika - Szczegóły kursu

Widok ze szczegółami (rysunek numer 20) pozwala nam na zapoznanie się z:

- Korzyściami płynącymi z całego przeprowadzanego szkolenia i warunki jakie musimy spełnić, aby do niego przystąpić
- Informacjami o czasie trwania, lokalizacji, prowadzącym itp.
- Ceną kursu
- Dostępnymi terminami

Jak można zauważyc, w chwili obecnej zapis kurs jest dla nas niedostępny, a to ze względu na fakt, że tylko zalogowani użytkownicy mogą tego dokonać, w przeciwnieństwie do rezerwacji eventów, na którą jesteśmy w stanie (podobnie jak dla kursów) dostać się z poziomu oferty i listy:

The screenshot shows a user interface for booking events. At the top, there's a header with a phone number (Tel. 123 456 121), an email address (wadventure@gmail.com), and links for 'Zaloguj się' (Log in) and 'Zarejestruj' (Register). Below the header, there are navigation links: Start, Oferta, E-learning, Moje kursy, Moje eventy, and O firmie.

WIECZÓR PANIEŃSKI
Sopot | Olecko
ONLINE / STACJONARNE

WIECZÓR KAWALERSKI
Sopot | Olecko
ONLINE / STACJONARNE

REJS WIDOKOWY
Sopot
ONLINE / STACJONARNE

Each event listing includes a small thumbnail image, the event name, location, and availability status. To the right of each listing is a summary of included items under 'W zestawie:' and a 'ZOBACZ WIĘCEJ' button. A tooltip for the 'ZOBACZ WIĘCEJ' button indicates it leads to a detailed view of the event.

Rysunek 21- Instrukcja użytkownika - Lista rejsów

Po przejściu w ‘Zobacz więcej’ ukazuje nam się panel jak z kursem:

The screenshot shows a detailed view of the 'Rejs widokowy' (Sightseeing Cruise) offer. At the top, there's a header with a phone number (Tel. 123 456 121), an email address (wadventure@gmail.com), and links for 'Zaloguj się' (Log in) and 'Zarejestruj' (Register). Below the header, there are navigation links: Start, Oferta, E-learning, Moje kursy, Moje eventy, and O firmie.

Rejs widokowy

A large image of a sailboat on the water is displayed. To the left of the image, there's a section titled 'Co zawiera oferta' (What does the offer include) with a bulleted list:

- Półtora godziny rejs dla uczestników (maksymalna liczba osób to 10)
- Atrakcyjne spędzenie czasu na wodzie
- Dostępna toaleta na pokładzie
- Możliwość włączenia własnej playlisty z głośników

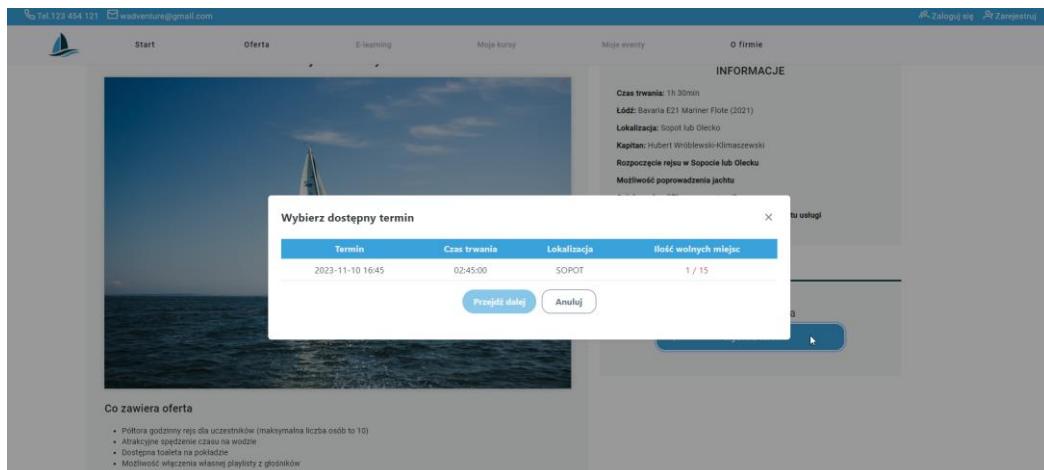
To the right of the image, there's a section titled 'INFORMACJE' with detailed information about the cruise:

- Czas trwania: 1h 30min
- Lódź: Bavaria E21 Mariner Flote (2021)
- Lokalizacja: Sopot lub Olecko
- Kapitan: Hubert Wróblewski-Klimaszewski
- Rozpoczęcie rejsu w Sopocie lub Olecku
- Możliwość poprowadzenia jachtu
- Opieka wykwalifikowanego sternika
- Kamizelki ratunkowe oraz ciepłe koce dla komfortu usługi

At the bottom right, there's a button labeled 'Wybierz termin' (Select date).

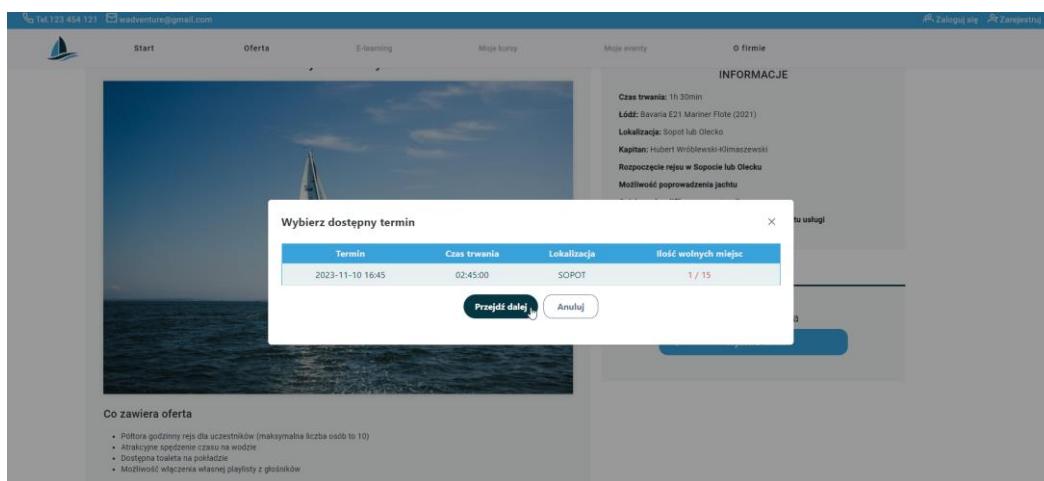
Rysunek 22 - Instrukcja użytkownika - Szczegóły rejsu

Tym razem jednak jesteśmy w stanie skorzystać z możliwości rezerwacji rejsu dzięki przyciskowi Wybierz termin, który otwiera okno dialogowe z wyborem konkretnego terminu:



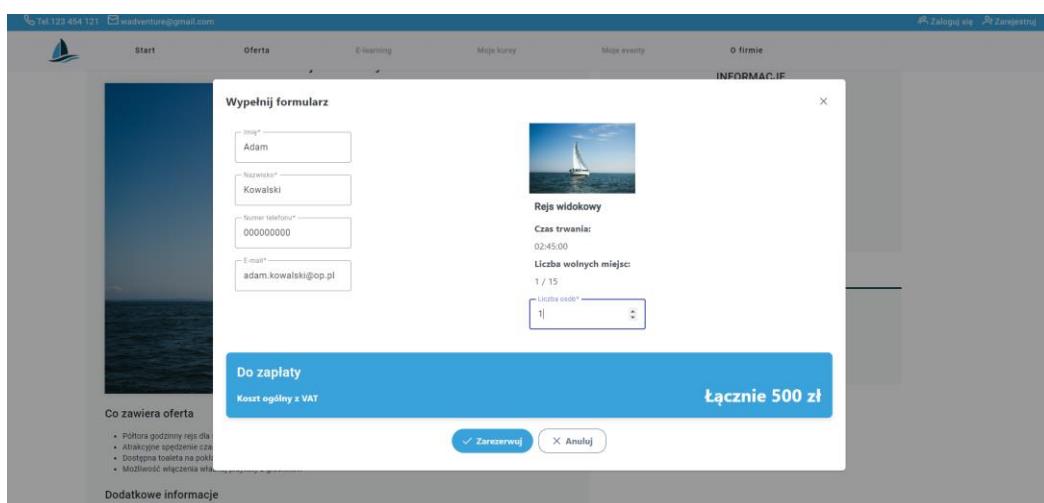
Rysunek 23 - Instrukcja użytkownika - Zapis na rejs - wybór terminu

Po wybraniu interesującego nas terminu, możemy użyć przycisku Przejdź dalej:



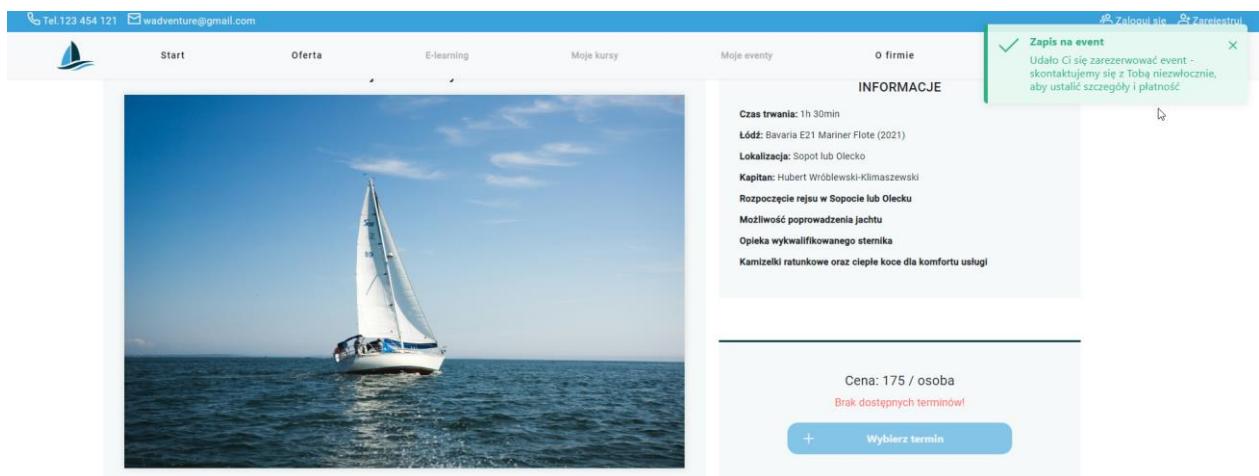
Rysunek 24 - Instrukcja użytkownika - Zapis na rejs - wybór terminu - aktywny

To prowadzi nas do formularza z rezerwacją:



Rysunek 25 - Instrukcja użytkownika - Zapis na rejs - formularz

Po jego pomyślnym wypełnieniu otrzymujemy komunikat od systemu:

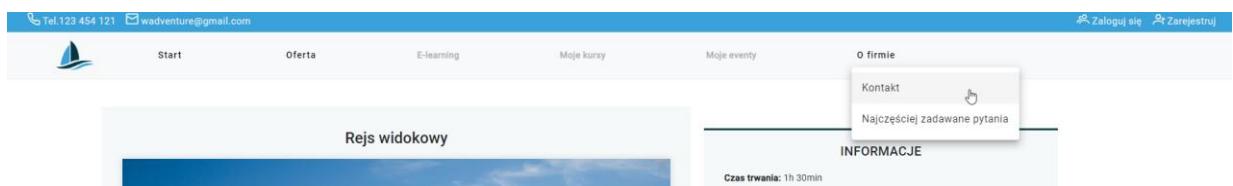


Rysunek 26 - Instrukcja użytkownika - Zapis na rejs - koniec procesu

Dzięki czemu udało nam się zarezerwować rejs na podany w formularzu email, który jest jednak w chwili obecnej nieopłacony i musimy zaczekać na kontakt administratora, w celu ustalenia szczegółów właśnie płatności i samego rejsu.

6.3 Zakładka O firmie

Klient korzystający ze strony, który jeszcze nie zdołał zarejestrować konta i zalogować się, ma dostęp do zakładki O firmie, która pozwala na przejrzenie informacji kontaktowych oraz rozwijalnej listy najczęściej zadawanych pytań:



Rysunek 27- Instrukcja użytkownika - Pasek nawigacyjny - Zakładka O firmie

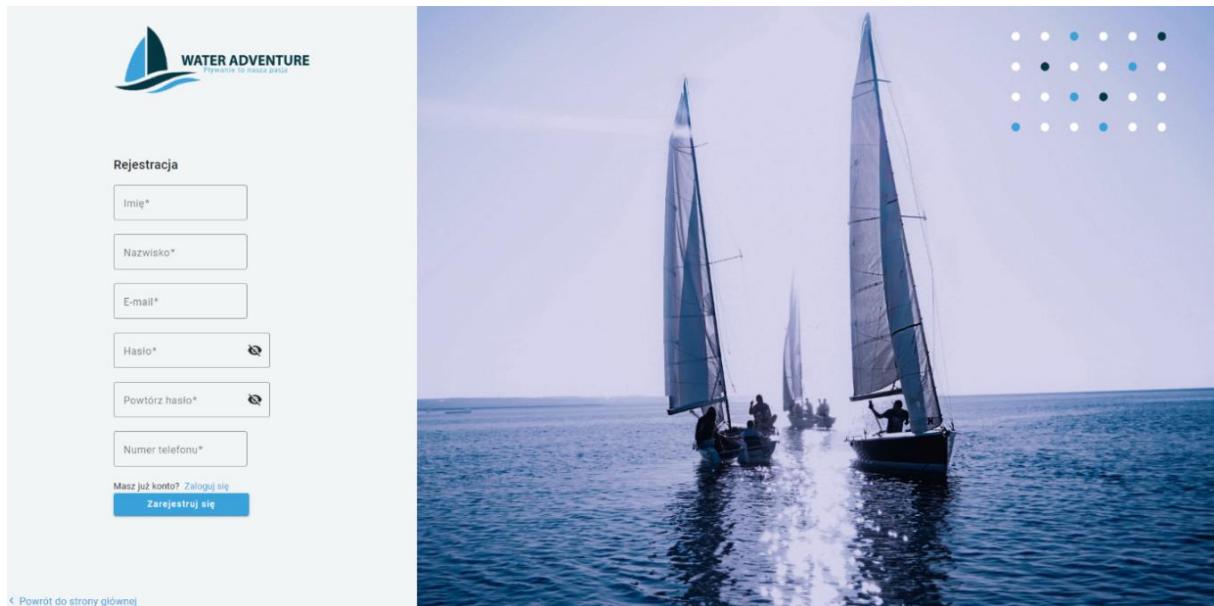
Rysunek 28- Instrukcja użytkownika - Strona kontaktowa

Rysunek 29- Instrukcja użytkownika - Strona kontaktowa – aktywna

Rysunek 30- Instrukcja użytkownika - Najczęściej zadawane pytania

6.4 Rejestracja

Widok zaprojektowany oczywiście w celu umożliwienia użytkownikom przystępnego założenia nowego konta, które pozwoli na pełne korzystanie z możliwości aplikacji



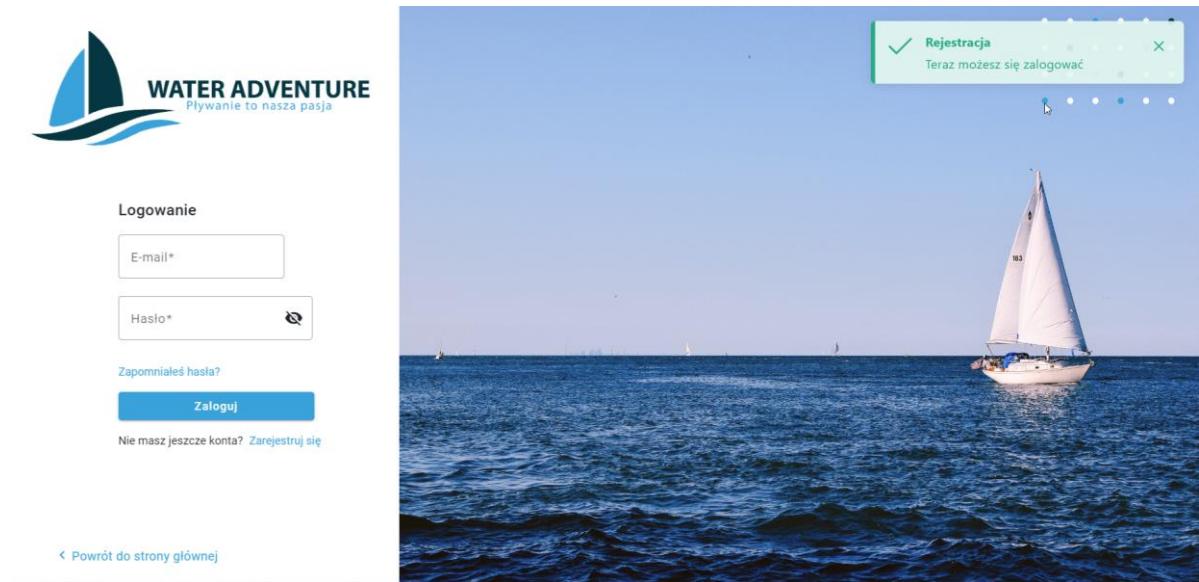
Rysunek 31- Instrukcja użytkownika - Rejestracja

Najważniejszą częścią tego komponentu jest formularz rejestracyjny, który składa się na walidowane indywidualnie pola:

A screenshot of the same registration form from Rysunek 31, but with validation errors applied to each field. The "Imię*" field has a red border and the error message "Musisz wypełnić pole!". The "Nazwisko*" field has a red border and the error message "Musisz wypełnić pole! E-mail*". The "E-Mail" field has a red border and the error message "Niepoprawny email! Hasło*". The "Hasło*" field has a red border and the error message "Pole musi zawierać min. 8 znaków! Powtóż hasło*". The "inneHasło" field has a red border and the error message "Hasła muszą być takie same! Numer telefonu*". The "AAA" field has a red border and the error message "Niepoprawny format". Below the form is a link "Masz już konto? Zaloguj się" and a blue "Zarejestruj się" button.

Rysunek 32- Instrukcja użytkownika - Rejestracja - formularz

Dopiero po wprowadzeniu poprawnych danych jesteśmy w stanie przejść przez proces rejestracji, po którym system wita nas na panelu logowania z następującym komunikatem:



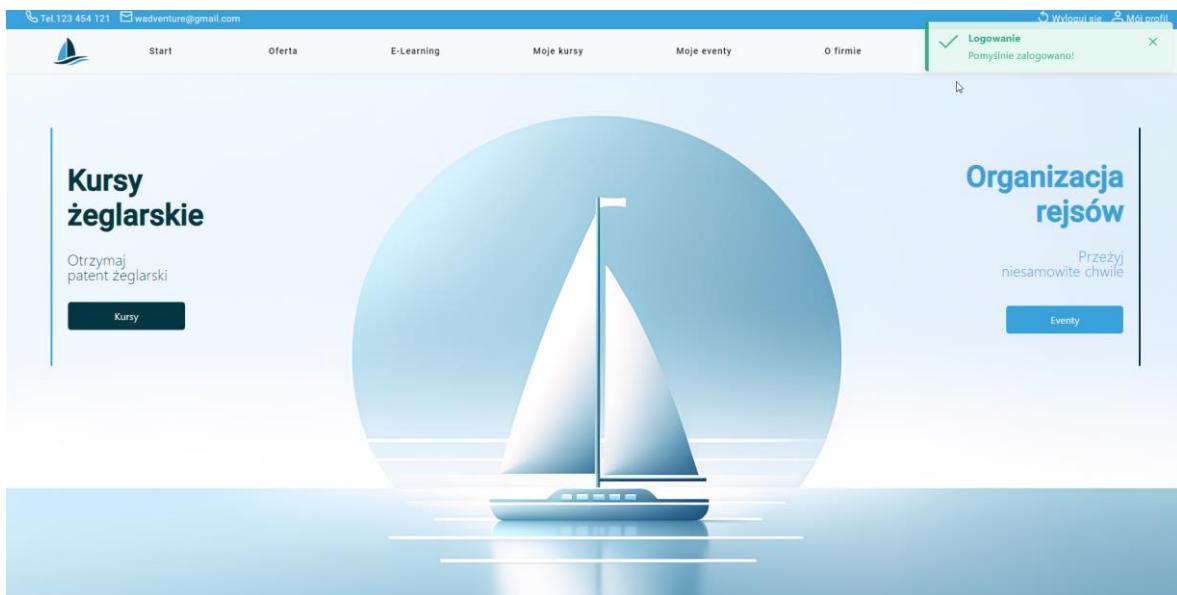
Rysunek 33- Instrukcja użytkownika - Logowanie

Proces logowania wygląda synonimycznie do rejestracji widocznej na rysunku numer 31, czyli wypełniamy widoczny formularz, tym razem danymi zarejestrowanego użytkownika, a następnie używamy przycisku Zaloguj:

A detailed view of the 'Logowanie' (Login) form. It shows the 'E-mail*' field containing 'igor.kawczynski@op.pl', the 'Hasło*' field with a redacted password, and the 'Zaloguj' button, which has a hand cursor icon indicating it is clickable. Below the form are links for 'Zapomniałeś hasła?' and 'Nie masz jeszcze konta? Zarejestruj się'.

Rysunek 34- Instrukcja użytkownika - Logowanie - formularz

Prowadzi to zalogowanego już użytkownika na stronę główną, gdzie widzimy, że jesteśmy w stanie w pełni korzystać z menu aplikacji:



Rysunek 35- Instrukcja użytkownika - Strona główna - Użytkownik zalogowany

Pozwala nam to na skorzystanie z szeregu nowych funkcjonalności.

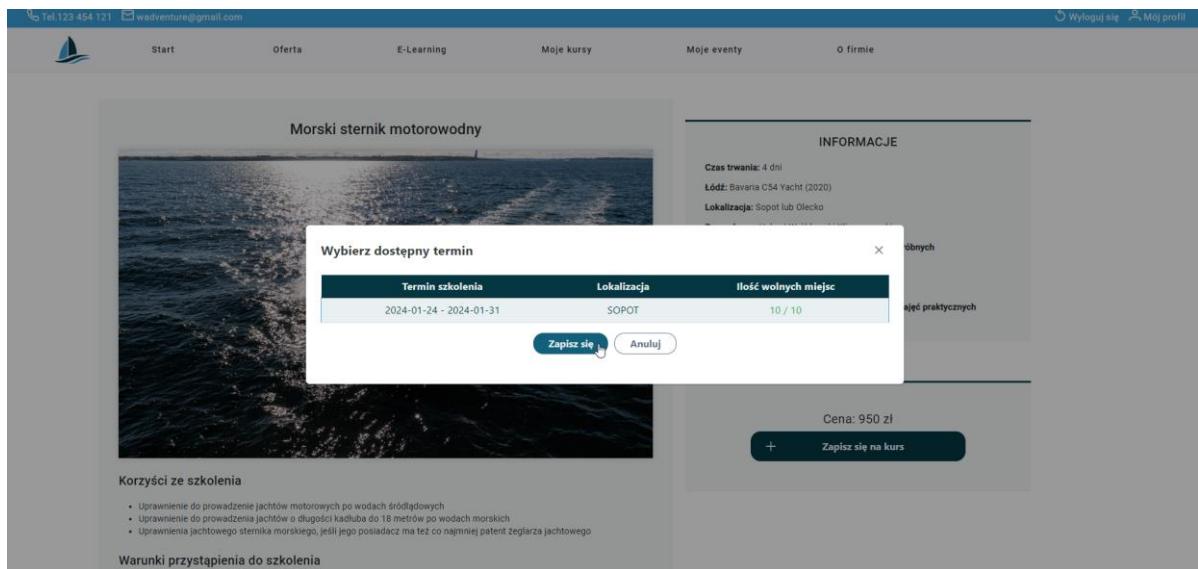
6.5 Zapis na kurs

Od teraz użytkownik jest w stanie pełnoprawnie skorzystać z opcji zapisu na kurs, dzięki czemu widok wygląda teraz inaczej:

A screenshot of a course registration page. At the top, there's a navigation bar with links for 'Start', 'Ofera', 'E-Learning', 'Moje kursy', 'Moje eventy', 'O firmie', 'Wydrukuj się', and 'Mój profil'. The main content area shows a boat on the water with the title 'Morski sternik motorowodny'. To the right, there's a 'INFORMACJE' section with details: 'Czas trwania: 4 dni', 'Łódź: Bavaria C54 Yacht (2020)', 'Lokalizacja: Sopot lub Olecko', 'Prowadzący: Hubert Wróblewski-Klimaszewski', 'Dostęp do bazy pytań (e-learning) i egzaminów próbnych', 'Egzamin w cenie kursu', 'Certyfikat po ukončení szkolenia', and 'Ubezpieczenie NNW i środki ratunkowe na czas zajęć praktycznych'. Below this is a price of 'Cena: 950 zł' and a 'Zapisz się na kurs' button. At the bottom, there are sections for 'Korzyści ze szkolenia' (listing benefits like permission to operate motor yachts in inland waters and permission to operate yachts up to 18 meters in marine waters) and 'Warunki przystąpienia do szkolenia' (listing requirements like having at least a basic sailing license).

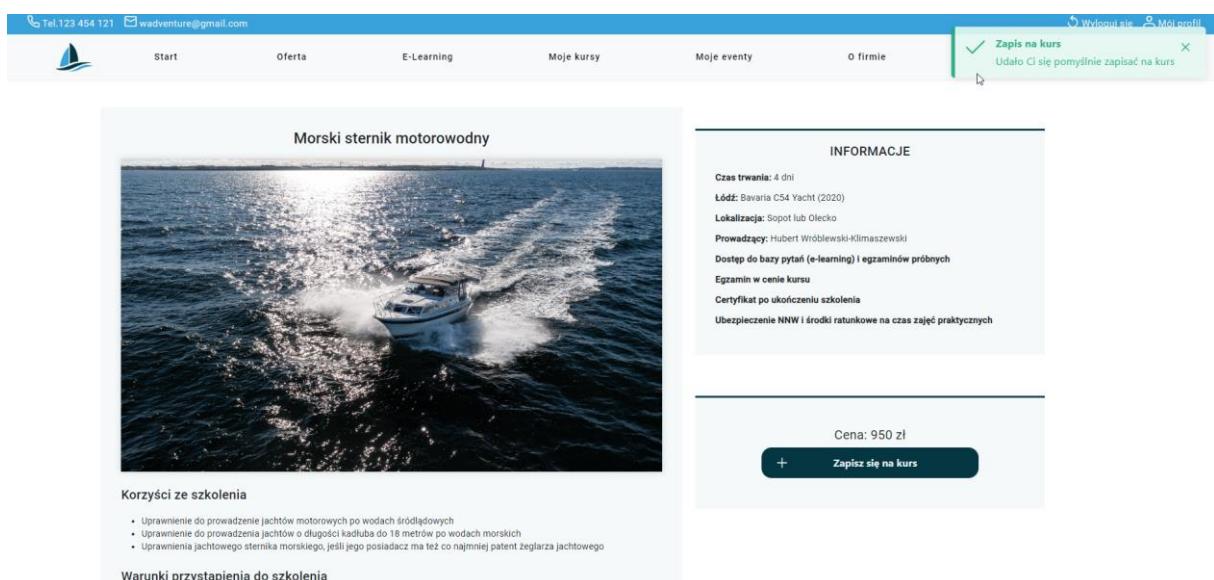
Rysunek 36 - Instrukcja użytkownika - Zapis na kurs - Użytkownik zalogowany

Nad przyciskiem do zapisu nie widnieje już czerwony napis, a ponadto jesteśmy w stanie z niego skorzystać, co prowadzi nas do okna dialogowego z wyborem terminu, podobnie jak rzecz miała się w rejsach:



Rysunek 37 - Instrukcja użytkownika - Zapis na kurs - wybór terminu

Po kliknięciu w przycisk **Zapisz się** widzimy:



Rysunek 38- Instrukcja użytkownika - Zapis na kurs - koniec procesu

Dzięki czemu jesteśmy pewni, że dla użytkownika został odpowiednio przypisany kurs, co możemy zobaczyć w następnym widoku.

6.6 Moje kursy

Przechodząc na omawianą zakładkę, będąc zalogowanym użytkownikiem, jesteśmy w stanie zobaczyć listę kursów, do których jesteśmy przypisani, co wygląda następująco:

The screenshot shows a course entry for a "Motorowodny sternik morski" (Motor Yacht Skipper) course in Sopot, running from January 24 to January 31, 2024. The course is marked as "Nieopłacony" (Unpaid). There are buttons for "Szczegóły" (Details) and "Anuluj" (Cancel).

The screenshot shows the user profile section with contact details for Sopot, Olecko, and Gdynia. It also displays two maps: one for Olecko showing the location of Przystan LOK and another for Sopot showing the location of Plac Zdrojowy. The Water Adventure logo is visible at the top.

Rysunek 39- Instrukcja użytkownika - Moje kursy - Użytkownik zalogowany

Taki kurs jesteśmy w stanie anulować, bądź przejść do jego szczegółów (Rysunek numer 35).

Kursy powinny jednak zostać w najbliższym czasie opłacone, po czym administrator serwisu zmienia jego status, a wtedy widok może wyglądać następująco:

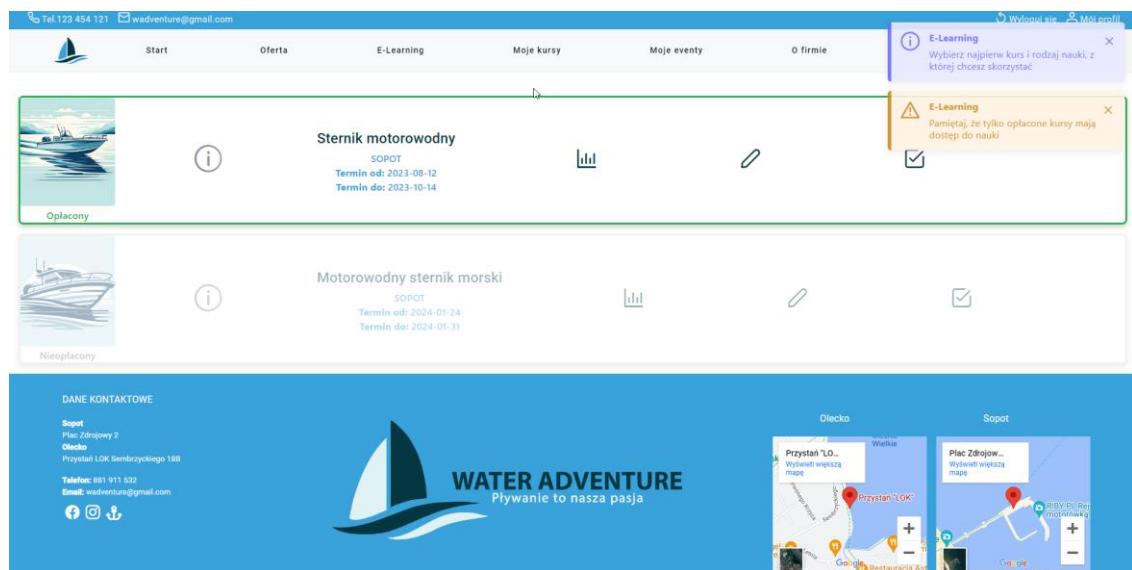
The screenshot shows a paid course entry for a "Sternik motorowodny" (Motor Yacht Skipper) course in Sopot, running from August 12 to October 14, 2023. It also shows a course entry for a "Motorowodny sternik morski" (Motor Yacht Skipper) course in Sopot, running from January 24 to January 31, 2024. The course in Sopot is marked as "Opłacony" (Paid). There are buttons for "Szczegóły" (Details), "Nauka" (Study), and "Anuluj" (Cancel). Below the courses, there is contact information and a map section.

Rysunek 40 - Instrukcja użytkownika - Moje kursy - Użytkownik zalogowany - kurs opłacony

Możemy teraz skorzystać z przycisku Nauka, który przekieruje nas na zakładkę E-Learning.

6.7 E-learning

Z poziomu paska nawigacyjnego (rysunek numer 13), bądź jak ukazano w zakładce Moje kursy (rysunek numer 39), jesteśmy w stanie przejść do komponentu z nauką zdalną, który prezentuje się tak:



Rysunek 41 - Instrukcja użytkownika - E-learning

Podczas pierwszego wejścia dostajemy dwa komunikaty informujące oraz listę kursów wraz z przyciskami do korzystania z nauki zdalnej (tylko dla kursów opłaconych).

Możemy następnie z poziomu opłaconego kursu przejść do:

1. Szczegółów kursu:



Rysunek 42 - Instrukcja użytkownika - E-learning – Szczegóły

Wyświetli nam się wówczas widok z rysunku numer 35.

2. Egzaminu próbnego:



Rysunek 43 - Instrukcja użytkownika - E-learning - Egzamin próbnny

System wygeneruje dla nas próbny test:

The screenshot shows a user interface for an e-learning platform. At the top, there is a navigation bar with links for 'Start', 'Oferta', 'E-Learning', 'Moje kursy', 'Moje eventy', and 'O firmie'. A user profile icon is also present. A sidebar on the right contains a purple box titled 'Egzamin próbny' with the text: 'Rozpoczynasz egzamin próbny. Po udzieleniu wszystkich odpowiedzi kliknij przycisk do ukonczenia egzaminu.' Below this, there is a question box for 'Pytanie numer 180' with three options: A, B, and C. Option A is selected. Navigation buttons for 'Poprzednie pytanie' and 'Następne pytanie' are visible. A 'Lista pytań' section shows a grid of numbered circles from 1 to 51. A timer at the bottom indicates 'Pozostały czas: 59:57'. A blue button labeled 'Zakończ egzamin' is located at the bottom right. The footer features a 'DANE KONTAKTOWE' section with 'Sopot' and 'Olecko' links, and a 'Sopot' link in the center.

Rysunek 44 - Instrukcja użytkownika - E-learning - Egzamin próbny - Rozwiązywanie egzaminu

W którym udzielamy odpowiedzi na listę 51 pytań pobranych z bazy danych, a następnie (lub po upłynięciu czasu) zostajemy przekierowani do widoku wyniku naszego testu:

The screenshot shows the results page of a sample exam. The top navigation bar and sidebar are identical to the previous screenshot. The main content area displays the title 'Egzamin próbny'. It shows the user's result: 'Twój wynik' (Your result) is 57%, indicated by a red '57%' and a red bar chart icon. A timer shows 'Pozostały czas' (Remaining time) as '58:22'. Below this, a checked checkbox icon indicates 'Poprawne odpowiedzi' (Correct answers) with the value '29/51'. At the bottom, there are two buttons: 'Moje wyniki' (My results) and 'Spróbuj ponownie' (Try again).

Rysunek 45 - Instrukcja użytkownika - Egzamin próbny - Wyniki egzaminu

Z którego możemy przejść do statystyk nauki (przycisk Moje wyniki) lub spróbować rozwiązać test ponownie (Spróbuj ponownie).

3. Nauki ogólnej:

Sternik motorowodny
SOPOT
Termin od: 2023-08-12
Termin do: 2023-10-14

Oplacony

Rysunek 46 - Instrukcja użytkownika - E-learning - Nauka ogólna

System na tym widoku wygeneruje dla nas losowe pytanie:

Tel. 123 454 121 wadventure@gmail.com Wyloguj się Mój profil

Start Oferta E-Learning Moje kursy Moje eventy O firmie

Pytanie numer 215
Osoby nie potrafiące pływać powinny

Dział
Wiadomości z zakresu ratownictwa wodnego

A: Mieć założone kamizelki ratunkowe i siedzieć pod pokładem
B: Muszą być przywiązanego do jachtu
C: Powinny zawsze mieć na sobie kamizelki ratunkowe i unikać siedzenia pod pokładem

Następne pytanie >

Sprawdź odpowiedź

DANE KONTAKTOWE
Sopot
Plac Zdrojowy 2
Olecko
Przystań LOK Sembryckiego 18B
Telefon: 881 911 532
Email: wadventure@gmail.com

WATER ADVENTURE
Pływanie to nasza paśnia

Olecko Sopot

Przystan "LO... Wyswietl większą mapę
Przystan "LOK"
RIBY DU Rej motorynka

Dane mapy / Warunki Zgłoś błąd w mapach

Rysunek 47 - Instrukcja użytkownika - E-learning - Nauka ogólna - Pytanie

Dla którego możemy sprawdzić poprawną odpowiedź wybierając A, B lub C oraz klikając w przycisk Sprawdź odpowiedź:

Tel. 123 454 121 wadventure@gmail.com Wyloguj się Mój profil

Start Oferta E-Learning Moje kursy Moje eventy O firmie

Pytanie numer 211
Podczas wykonywania manewru "człowiek za burtą" należy

Dział
Wiadomości z zakresu ratownictwa wodnego

A: cały czas obserwować jego pozycję
B: skoczyć za nim
C: pomagać człowiekowi na do widzenia

Następne pytanie >

Sprawdź odpowiedź

DANE KONTAKTOWE
Sopot
Plac Zdrojowy 2
Olecko
Przystań LOK Sembryckiego 18B
Telefon: 881 911 532
Email: wadventure@gmail.com

WATER ADVENTURE
Pływanie to nasza paśnia

Olecko Sopot

Przystan "LO... Wyswietl większą mapę
Przystan "LOK"
RIBY DU Rej motorynka

Dane mapy / Warunki Zgłoś błąd w mapach

Rysunek 48 - Instrukcja użytkownika - E-learning - Nauka ogólna - Poprawna odpowiedź

4. Statystyki nauki:

Rysunek 49 - Instrukcja użytkownika - E-learning - Statystyki nauki

W tej zakładce system analizuje nasze wyniki z nauki ogólnej oraz próbnego egzaminu w celu wyświetlenia ich użytkownikowi:

Kategoria	Wartość
Poprawne odpowiedzi	~8
Wszystkie odpowiedzi	35

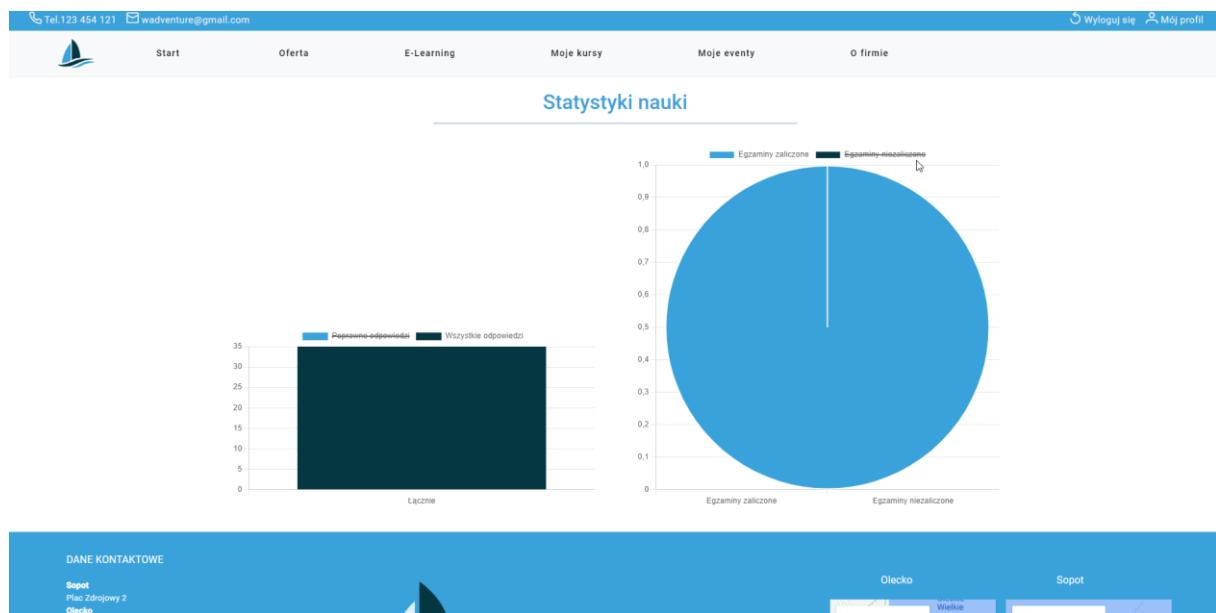
Rysunek 50 - Instrukcja użytkownika - E-learning - Statystyki nauki - Widok ogólny

Aplikacja przedstawia nam dwa wykresy, jeden uwzględniający wszystkie udzielone oraz poprawne odpowiedzi, drugi natomiast dotyczy się rozwiązań egzaminów próbnych i procentu ich zaliczenia. Możemy najechać na każdy wykres, co pozwoli nam na odczytanie informacji w formie tekstu:

Kategoria	Wartość
Poprawne odpowiedzi	~8
Wszystkie odpowiedzi	35

Rysunek 51 - Instrukcja użytkownika - E-learning - Statystyki nauki - Tekst informacyjny

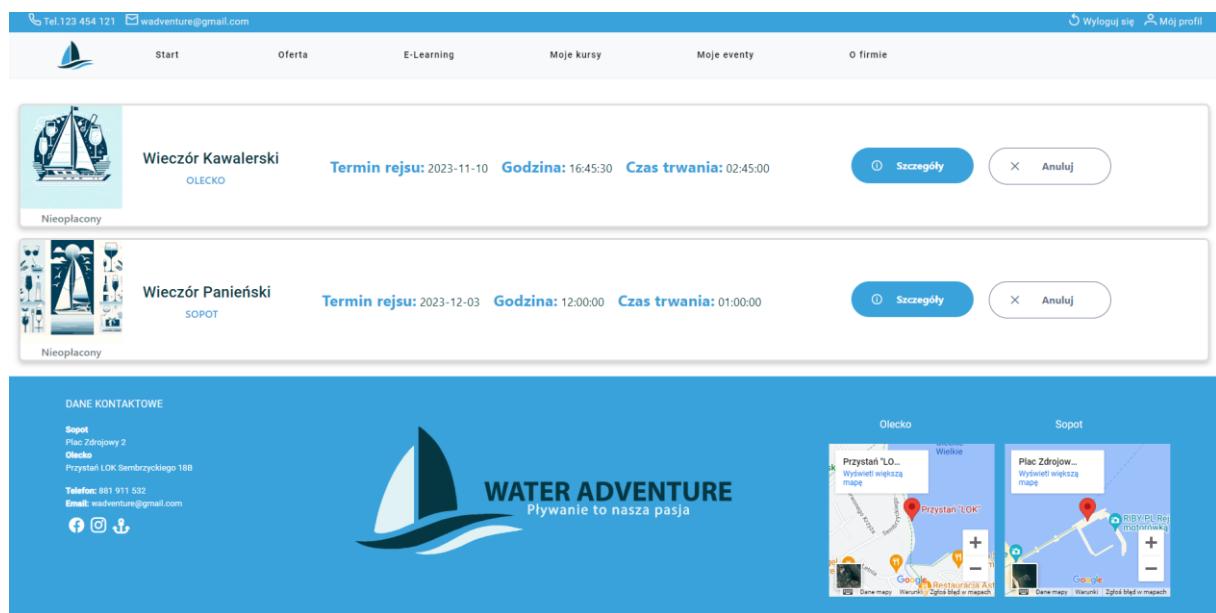
Aplikacja umożliwia także wybranie zbioru danych, który chcemy wyświetlić, a drugi ukryć:



Rysunek 52- Instrukcja użytkownika - E-learning - Statystyki nauki - Wyświetlenie konkretnego zbioru danych

6.8 Moje eventy

Widok moich eventów przedstawia nam listę zarezerwowanych rejsów, które są przypisane do aktualnie używanego na koncie użytkownika emaila:



Rysunek 53 - Instrukcja użytkownika - Moje eventy

Podobnie jak w przypadku kursów, tak i tutaj jesteśmy w stanie wycofać nasze uczestnictwo w wydarzeniu, wymaga to jednak odpowiedniego odstępu czasowego pomiędzy dniem wypłynięcia, a dniem aktualnym i z tego faktu możemy wycofać się z eventów tylko, gdy rozpoczyna on się co najmniej 48h licząc od teraz, inaczej otrzymamy komunikat:

Rysunek 54 - Instrukcja użytkownika - Moje eventy - Nieprawidłowe anulowanie

6.9 Mój profil

Strona "Mój profil" jest osobistą przestrzenią użytkownika w aplikacji, zapewniającą dostęp do danych konta i umożliwiającą ich edycję. Jest to element aplikacji webowej, pozwalający na personalizację doświadczenia użytkownika i zarządzanie własnymi danymi.

Rysunek 55 - Instrukcja użytkownika - Mój profil

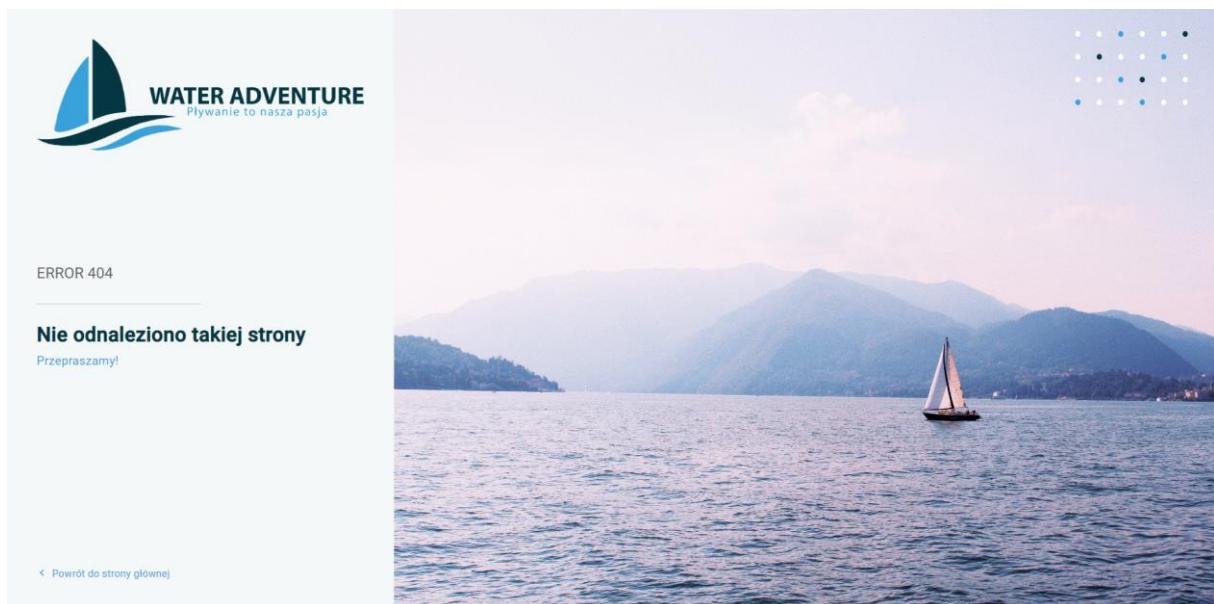
Z lewej strony panelu mamy dostęp do statystyk pokazujących jak długo nasze konto jest w systemie aplikacji oraz liczbę wykupionych i zaliczonych egzaminów.

Z prawej strony natomiast widzimy formularz, który nie musimy uzupełniać w całości – wystarczy, że wypełnimy jedno pole, które chcemy dla naszego konta zmienić oraz hasło, które weryfikuje, czy na pewno odpowiednia osoba dokonuje edycji.

Po kliknięciu w przycisk Zapisz zmiany konto powinno przejść modyfikację, a system wprowadzone zmiany zapisze do bazy danych.

6.10 Strona 404

Aplikacja obsługuje także sytuacje, w których użytkownik przejdzie na adres strony w zakresie domeny aplikacji, która nie istnieje, wyświetla się wówczas użytkownikowi widok obsługujący popularny błąd typu 404 i umożliwiający wygodny powrót na stronę główną:



Rysunek 56 - Instrukcja użytkownika - Strona 404

Strony obsługujące błąd 404 są używane w celu poprawy doświadczenia użytkownika i dostarczenia mu jasnej informacji, gdy próbuje odwiedzić nieistniejący na serwerze klienta adres URL.

7. Testy

Testy stanowią istotny element omawianej aplikacji sporządzonej dla pracy dyplomowej, gdyż zapewniają większą niezawodność tworzonego systemu.

W celach automatyzacji testów stworzone zostały testy jednostkowe zarówno po stronie backendu i frontendu. Testy jednostkowe są niezbędnym narzędziem w arsenale każdego programisty, które służą do weryfikacji działania najmniejszych, izolowanych części aplikacji - tzw. jednostek. Stanowią one pierwszą linię obrony przed błędami w kodzie, pozwalając na szybkie wykrywanie i naprawianie problemów na wczesnym etapie cyklu życia oprogramowania. Przy ich pomocy można sprawdzić, czy poszczególne komponenty działają zgodnie z oczekiwaniemi zarówno w izolacji, jak i w interakcji z innymi elementami systemu.

Testy jednostkowe są także nieodzownym elementem tak zwanego Continuous Integration (CI) i Continuous Delivery (CD), gdzie każde wdrożenie nowego kodu programistycznego jest automatycznie testowane. To pozwala na szybką identyfikację oraz naprawę błędów, zanim jeszcze staną się one częścią głównej gałęzi rozwoju aplikacji. W przeciwieństwie do testów manualnych, które są czasochłonne, podatne na błędy i wymagają dodatkowych zasobów, testy jednostkowe można szybko uruchomić, co zapewnia konsystencję w procesie testowania, i tym samym można je określić inwestycją w przyszłość w kontekście programowania aplikacji webowych.

7.1 Testy backendowe

Na backendzie, testy jednostkowe są stworzone przy użyciu frameworka JUnit. Jest to jedno z najpopularniejszych narzędzi w ekosystemie Javy, pozwalające na szybkie i skuteczne tworzenie testów. JUnit charakteryzuje się prostotą w konfiguracji oraz bogatym zestawem asercji, które pomagają w weryfikacji wyników testów. W połączeniu z Mockito, biblioteką do tworzenia obiektów, które symulują byty rzeczywiste, byłem w stanie symulować zależności i zachowania zewnętrznych komponentów systemu, co jest szczególnie przydatne w przypadku testowania logiki biznesowej niezależnie od warstw serwisowych, dostępu do bazy danych czy zewnętrznych API.

Najistotniejsze sektory serwisu backendowego, które zostały przetestowane:

- **Edycja kursu** – Odpowiada za przetestowanie aktualizacji informacji o kursie, testuję pomyślnie przeprowadzoną modyfikację (wszystkie dane poprawne), a także scenariusz, w którym wprowadzone dane nie są poprawne.
- **Edycja rejsu** – Test synonimiczny do testu powyższego, który sprawdza wszystkie możliwe scenariusze przy edycji rejsu.
- **Rejestracja użytkownika** – Test weryfikujący scenariusz poprawnej rejestracji, w którym sprawdzane jest, czy stworzony użytkownik został poprawnie zapisany w bazie danych i czy jego hasło zostało pomyślnie zahaszowane algorytmem BCrypt⁸.
- **Logowanie** – Test sprawdza scenariusze, w których logowanie przechodzi pomyślnie i tworzony zostaje token JWT zapisywany do ciasteczek w sesji użytkownika oraz taki, w którym logowanie wyrzuca wyjątek błędnie wprowadzonych danych przez co nie zostaje stworzony token JWT.
- **Filtrowanie kontekstu zabezpieczeń** – Sprawdza, czy użytkownik po poprawnie przeprowadzonym procesie nadania autoryzacji ma przypisany poprawny token JWT podpisany jego emailem oraz czy wysyłane przez niego żądania do serwera są poprawnie filtrowane w kontekście Spring Security, a także scenariusz, w którym użytkownik nie otrzymuje tokenu i uprawnień, a następnie próbuje wysłać żadanie, które takich zasobów wymaga.
- **Zapis na kurs** – Podobnie jak w przypadku edycji kursów i rejsów, test obejmuje scenariusze pomyślnego zapisu na kurs oraz zapisu, który prowadzi do wyrzucenia wyjątków.
- **Anulowanie zapisu na kurs** – Sprawdza czy metoda usunięcia relacji pomiędzy użytkownikiem, a kursem działa poprawnie i czy wszystkie elementy z bazy danych są usuwane, a także przypadek, w którym kursant próbuje anulować opłacony już kurs.

⁸ BCrypt - Jeden z najbardziej popularnych algorytmów haszowania, który jest szeroko stosowany w celu zabezpieczenia poufnych informacji przed atakami hackerskimi. Jest to algorytm oparty na funkcji haszującej, która konwertuje wskazaną ilość danych wejściowych na skrót o stałej długości.

7.2 Testy frontendowe

W przypadku frontendu, testy jednostkowe zostały zrealizowane dzięki frameworkowi Jasmine, który służy do opisywania testów i zapewnienia asercji w przyjaznej dla użytkownika składni. Jasmine umożliwia testowanie funkcjonalności kodu bez potrzeby interakcji z przeglądarką, co znacząco przyspiesza proces testowania. Dla dopełnienia frameworka Jasmine użyłem także Karmy, czyli tak zwanego test runnera, który umożliwia uruchomienie testów napisanych w Jasmine na rzeczywistej przeglądarce, co jest istotne dla prawdziwego, wiernego odwzorowania środowiska uruchamianego dla testów.

Dla frontendu zostało przetestowane przede wszystkim:

- **Edycja oraz usunięcie kursu** (wszystkie możliwe scenariusze)
- **Edycja oraz usunięcie rejsu** (prowadzi do wyrzucenia wyjątków)
- **Edycja użytkownika** (sprawdza, czy sesja użytkownika została zaktualizowana nowymi danymi)
- **Nauka ogólna** (sprawdza, czy inicjalizacja i udzielenie poprawnej/błędnej odpowiedzi pytania odbywa się poprawnie)
- **Egzamin próbny** (testuje, czy czas jest odliczany poprawnie oraz czy po skończeniu egzaminu jego rezultat jest wprowadzany do statystyk nauki)

7.3 Testy osób trzecich

Oprócz testów wprowadzanych bezpośrednio do kodu, aplikacja została także udostępniona kilkunastu osobom nie związanym stricte ze środowiskiem programistycznym, co pozwoliło na momentami bardziej ‘naturalne’ podejście do interfejsu. Testerzy ci, nieobciążeni wiedzą o oczekiwanych ścieżkach dojścia do konkretnych funkcjonalności aplikacji, byli w stanie niejednokrotnie dać cenne wskazówki, które mogły zostać przeoczone na etapie implementacji lub testowania automatycznego i finalnie przyłożyły się do poprawienia ogólnych odczuć płynących z użytkowania niniejszej aplikacji.

8. Podsumowanie

8.1 Wnioski

Głównym celem aplikacji było umożliwienie użytkownikom łatwego zapisu na kursy żeglarskie, rejsy, oraz dostęp do materiałów edukacyjnych – wszystko w przystępnej i bezpiecznej formie. Te założenia udało się spełnić, co pozwoliło na stworzenie wszechstronnej platformy, z której mogłyby w przyszłości korzystać firmy z branży żeglarskiej. Stworzenie niniejszej aplikacji webowej stanowiło kompleksowe przedsięwzięcie, które przyniosło szereg wartościowych wniosków zarówno z perspektywy technologicznej, jak i użytkowej:

- 1. Rozwinięcie umiejętności technicznych:** Tworzenie tej aplikacji wymagało głębszego zapoznania się z różnymi technologiami i narzędziami, takimi jak HTML, CSS, TypeScript oraz frameworkami do budowy interfejsów użytkownika lub obsługi logiki biznesowej jak Angular i Spring Boot. To wszystko pomogło mi poszerzyć zakres moich umiejętności programistycznych.
- 2. Doświadczenie w projektowaniu aplikacji:** Praca nad tym projektem umożliwiła mi także zdobycie praktycznego doświadczenia w projektowaniu i tworzeniu aplikacji webowych od podstaw. Rozwinąłem swoje umiejętności od planowania po wdrożenie i testowanie.
- 3. Zrozumienie potrzeb użytkowników:** Praca nad niniejszą aplikacją pozwoliła mi lepiej zrozumieć potrzeby i oczekiwania użytkowników oraz dostosować rozwiązania techniczne do ich wymagań. To kluczowy aspekt w procesie tworzenia użytecznych i efektywnych aplikacji.

8.2 Możliwości rozwoju

Pomimo spełnienia priorytetowych celów, aplikacja jest gotowa na dalszy rozwój i rozszerzenie swoich funkcjonalności. Dzięki wysokiej skalowalności systemu, są obszary, które chciałbym w niedalekiej przyszłości rozbudować, aby jeszcze bardziej zwiększyć atrakcyjność i użyteczność aplikacji. Oto niektóre z proponowanych usprawnień i rozszerzeń, które zostaną wprowadzone w kolejnych etapach rozwoju aplikacji:

- **Pełniesze zabezpieczenie użytkownika pod kątem Spring Security:** Rozszerzenie zabezpieczeń w aplikacji jest niezwykle ważne. Dalsza rozbudowa to przede wszystkim autoryzacja dwuskładnikowa, token JWT z odświeżeniem oraz możliwość odzyskiwania hasła.
- **Zabezpieczone płatności online:** Integracja z gotowym API do obsługi płatności i banków jest pomysłem, który także zostanie wprowadzony w bliskim czasie. To uczyni proces płatności dla rejsów i kursów bardziej wygodnym i bezpiecznym.
- **Rozszerzone opisy kursów i rejsów:** Dodanie bardziej zróżnicowanych i szczegółowych opisów dla kursów i rejsów może pomóc użytkownikom lepiej zrozumieć ofertę.
- **Rozszerzony moduł e-learningowy:** Rozwinięcie modułu e-learningowego o naukę na podstawie wybranego działu żeglarskiego i bardziej rozbudowaną bazę pytań z pewnością zwiększy wartość edukacyjną. To może przyciągnąć więcej użytkowników i pomóc w poprawie wyników z egzaminów.
- **Rozszerzona analiza statystyk nauki:** Dodanie kolejnego wykresu do statystyk może pomóc użytkownikom śledzić swój postęp i identyfikować obszary do poprawy w lepszym stopniu. Taki wykres mógłby uwzględnić omawianie w rozszerzeniu całego modułu kategorie i sprawdzać jak użytkownik radził sobie w poszczególnych dziedzinach nauki.
- **Pełnoprawny formularz kontaktowy:** Formularz kontaktowy zamiast strony zawierającej informacje kontaktowe ułatwiłby komunikację między użytkownikami, a administratorem aplikacji.
- **Generowanie certyfikatów w formacie PDF:** Możliwość generowania certyfikatów po zakończeniu egzaminu w celu usprawnienia procesu otrzymania pełnoprawnego patentu.

Największą możliwością jest jednak rozwój osobisty w kontekście omawianej aplikacji, a także przyszłych projektów informatycznych. Niniejsza praca dyplomowa przyczyniła się w znacznym stopniu do zdobycia jeszcze większego zaangażowania w kategorii budowania aplikacji webowych, co stanowi największe świadectwo pracy.

Bibliografia

Literatura

1. Robert C. Martin - Czysty kod. Podręcznik dobrego programisty
Wydawnictwo Helion, 2015
2. Deborah Kurata - Doing Web Development: Client-Side Techniques
Wydawnictwo Apress, 2013
3. Cay S. Horstmann – Java 8 – Przewodnik doświadczonego programisty
Wydawnictwo Helion, 2015
4. Cay S. Horstmann, Gary Cornell – Java, techniki zaawansowane, wydanie IX
Wydawnictwo Helion, 2020
5. Walls Craig – Spring Boot in Action
Wydawnictwo Manning Publications, 2015
6. Vladimir Khorikov - Testy jednostkowe. Zasady, praktyki i wzorce
Wydawnictwo Helion, 2020
7. Adam Freeman - Angular. Profesjonalne techniki programowania. Wydanie IV
Wydawnictwo Helion, 2021
8. Adam Freeman - TypeScript 4. Od początkującego do profesjonalisty
Wydawnictwo Helion, 2022
9. Eran Kinsbruner - Testowanie aplikacji dla programistów frontendowych
Wydawnictwo Helion, 2023
10. Titus Winters, Tom Mansreck, Hyrum Wright - Inżynieria oprogramowania według Google. Czego warto się nauczyć o tworzeniu oprogramowania
Wydawnictwo Helion, 2023

11. Ford Neal, Richards Mark, Pramod Sadalage, Zhamak Dehghani - Złożone zagadnienia architektury oprogramowania. Jak analizować kompromisy i podejmować trudne decyzje Wydawnictwo Helion, 2023

Dokumentacja wykorzystanych technologii

- **PostgreSQL** - <https://www.postgresql.org/docs/>
- **Flyway** - <https://documentation.red-gate.com/flyway>
- **Java** - <https://docs.oracle.com/en/java/>
- **Spring Boot** - <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>
- **Spring Security** - <https://docs.spring.io/spring-security/reference/index.html>
- **Java Persistence API** - <https://javadoc.io/doc/javax.persistence/javax.persistence-api/latest/index.html>
- **Hibernate** - <https://hibernate.org/orm/documentation/6.2>
- **JUnit** - <https://junit.org/junit5/docs/current/user-guide>
- **Mockito** - <https://javadoc.io/doc/org.mockito/mockito-core/latest/org/mockito/Mockito.html>
- **HTML** - <https://developer.mozilla.org/en-US/docs/Web/HTML>
- **CSS** - <https://developer.mozilla.org/en-US/docs/Web/CSS>
- **JavaScript** - <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- **TypeScript** - <https://www.typescriptlang.org/docs/>
- **NPM** - <https://docs.npmjs.com/>
- **Angular** - <https://angular.io/docs>
- **Jasmine** - <https://jasmine.github.io/>
- **Bootstrap** - <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- **PrimeNG** - <https://primeng.org/installation>
- **Angular Material** - <https://material.angular.io/guide/getting-started>

Spis diagramów

Diagram 1 - Przypadków Użycia - Ogólny	21
Diagram 2 - Przypadków Użycia - PU Wyświetlenie informacji ze strony.....	22
Diagram 3 - Przypadków Użycia - PU Zarządzanie rejsami.....	22
Diagram 4 - Przypadków Użycia - PU Zarządzanie kursami	23
Diagram 5 - Przypadków Użycia - PU E-Learning.....	23
Diagram 6 - Przypadków Użycia - PU Zarządzanie profilem.....	24

Spis tabel

Tabela 1 - Scenariusz Przypadku Użycia 1.1 - Wyświetlenie dostępnych rejsów.....	25
Tabela 2 - Scenariusz Przypadku Użycia 1.2 - Wyświetlenie dostępnych kursów.....	26
Tabela 3 - Scenariusz Przypadku Użycia 1.3 - Wyświetlenie informacji kontaktowych	27
Tabela 4 - Scenariusz Przypadku Użycia 1.4 - Wyświetlenie najczęściej zadawanych pytań o kursy i rejsy	27
Tabela 5 - Scenariusz Przypadku Użycia 2.1 - Rezerwacja rejsu	29
Tabela 6 - Scenariusz Przypadku Użycia 2.2 - Podgląd zbliżających się rejsów użytkownika	29
Tabela 7 - Scenariusz Przypadku Użycia 2.3 - Wycofanie się z rejsu	30
Tabela 8 - Scenariusz Przypadku Użycia 3.1 - Zapis na kurs	31
Tabela 9 - Scenariusz Przypadku Użycia 3.2 - Podgląd zbliżających się kursów użytkownika	31
Tabela 10 - Scenariusz Przypadku Użycia 3.3 - Wycofanie się z kursu	32
Tabela 11 - Scenariusz Przypadku Użycia 4.1 - Wygenerowanie egzaminu próbnego.....	33
Tabela 12 - Scenariusz Przypadku Użycia 4.2 - Rozwiązywanie egzaminu próbnego	33
Tabela 13 - Scenariusz Przypadku Użycia 4.3 - Nauka ogólna	34
Tabela 14 - Scenariusz Przypadku Użycia 4.4 - Statystyki nauki.....	35
Tabela 15 - Scenariusz Przypadku Użycia 5.1 - Rejestracja konta	36
Tabela 16 - Scenariusz Przypadku Użycia 5.2 - Logowanie.....	37
Tabela 17 - Scenariusz Przypadku Użycia 5.3 - Wylogowanie	37
Tabela 18 - Scenariusz Przypadku Użycia 5.4 - Edycja profilu.....	38
Tabela 19 - Scenariusz Przypadku Użycia 5.5 - Wyświetlenie statystyk profilu.....	39
Tabela 20 - Opis pól i relacji encji Questions	42
Tabela 21 - Opis pól i relacji encji Events	42
Tabela 22 - Opis pól i relacji encji Participant events.....	43
Tabela 23 - Opis pól i relacji encji Participants	44
Tabela 24 - Opis pól i relacji encji Participant courses	44
Tabela 25 - Opis pól i relacji encji Courses	45
Tabela 26 - Opis pól i relacji encji General learning	45
Tabela 27 - Opis pól i relacji encji Category learning	46
Tabela 28 - Opis pól i relacji encji Trial exam.....	46

Spis ilustracji

Rysunek 1 - Schemat bazy danych.....	40
Rysunek 2 - Ogólna struktura backendu aplikacji	47
Rysunek 3 - Struktura pakietu models	48
Rysunek 4 - Struktura pakietu services	50
Rysunek 5 - Struktura pakietu controllers	51
Rysunek 6 - Struktura pakietu exceptions	52
Rysunek 7 - Struktura pakietu config.....	53
Rysunek 8 - Struktura pakietu migration	53
Rysunek 9 - Struktura pakietu test	54
Rysunek 10 - Logotyp aplikacji	58
Rysunek 11 - Instrukcja użytkownika - Strona główna	59
Rysunek 12 - Instrukcja użytkownika - Pasek nawigacyjny	59
Rysunek 13 - Instrukcja użytkownika - Strona główna - Część druga	60
Rysunek 14 - Instrukcja użytkownika - Strona główna - Część trzecia.....	61
Rysunek 15 - Instrukcja użytkownika - Strona główna - część czwarta	61
Rysunek 16 - Instrukcja użytkownika - Stopka	62
Rysunek 17 - Instrukcja użytkownika - Pasek nawigacyjny - Zakładka Oferta	62
Rysunek 18 - Instrukcja użytkownika - Lista kursów	63
Rysunek 19 - Instrukcja użytkownika - Szczegóły kursu	63
Rysunek 20 - Instrukcja użytkownika - Lista rejsów	64
Rysunek 21 - Instrukcja użytkownika - Szczegóły rejsu	64
Rysunek 22 - Instrukcja użytkownika - Zapis na rejs - wybór terminu	65
Rysunek 23 - Instrukcja użytkownika - Zapis na rejs - wybór terminu - aktywny	65
Rysunek 24 - Instrukcja użytkownika - Zapis na rejs - formularz	65
Rysunek 25 - Instrukcja użytkownika - Zapis na rejs - koniec procesu.....	66
Rysunek 26 - Instrukcja użytkownika - Pasek nawigacyjny - Zakładka O firmie	66
Rysunek 27 - Instrukcja użytkownika - Strona kontaktowa	67
Rysunek 28 - Instrukcja użytkownika - Strona kontaktowa - aktywna.....	67
Rysunek 29 - Instrukcja użytkownika - Najczęściej zadawane pytania.....	67
Rysunek 30 - Instrukcja użytkownika - Rejestracja.....	68
Rysunek 31 - Instrukcja użytkownika - Rejestracja - formularz.....	68
Rysunek 32 - Instrukcja użytkownika - Logowanie.....	69

Rysunek 33 - Instrukcja użytkownika - Logowanie - formularz.....	69
Rysunek 34 - Instrukcja użytkownika - Strona główna - Użytkownik zalogowany	70
Rysunek 35 - Instrukcja użytkownika - Zapis na kurs - Użytkownik zalogowany.....	70
Rysunek 36 - Instrukcja użytkownika - Zapis na kurs - wybór terminu	71
Rysunek 37 - Instrukcja użytkownika - Zapis na kurs - koniec procesu.....	71
Rysunek 38 - Instrukcja użytkownika - Moje kursy - Użytkownik zalogowany	72
Rysunek 39 - Instrukcja użytkownika - Moje kursy - Użytkownik zalogowany - kurs opłacony	72
Rysunek 40 - Instrukcja użytkownika - E-learning	73
Rysunek 41 - Instrukcja użytkownika - E-learning – Szczegóły	73
Rysunek 42 - Instrukcja użytkownika - E-learning - Egzamin próbny	73
Rysunek 43 - Instrukcja użytkownika - E-learning - Egzamin próbny - Rozwiążanie egzaminu	74
Rysunek 44 - Instrukcja użytkownika - Egzamin próbny - Wyniki egzaminu	74
Rysunek 45 - Instrukcja użytkownika - E-learning - Nauka ogólna	75
Rysunek 46 - Instrukcja użytkownika - E-learning - Nauka ogólna - Pytanie.....	75
Rysunek 47 - Instrukcja użytkownika - E-learning - Nauka ogólna - Poprawna odpowiedź ..	75
Rysunek 48 - Instrukcja użytkownika - E-learning - Statystyki nauki	76
Rysunek 49 - Instrukcja użytkownika - E-learning - Statystyki nauki - Widok ogólny	76
Rysunek 50 - Instrukcja użytkownika - E-learning - Statystyki nauki - Tekst informacyjny ..	76
Rysunek 51 - Instrukcja użytkownika - E-learning - Statystyki nauki - Wyświetlenie konkretnego zbioru danych	77
Rysunek 52 - Instrukcja użytkownika - Moje eventy	77
Rysunek 53 - Instrukcja użytkownika - Moje eventy - Nieprawidłowe anulowanie	78
Rysunek 54 - Instrukcja użytkownika - Mój profil	78
Rysunek 55 - Instrukcja użytkownika - Strona 404	79