

# **Zdrowy Student**

Dokumentacja projektu realizowanego w ramach  
przedmiotu Zaawansowane Technologie Internetowe

Kinga Miszczak

WFiIS IS

# 1. Cele projektu

Celem aplikacji „Zdrowy student” jest codzienne monitorowanie stanu zdrowia studenta poprzez odpowiednie statystyki i subiektywnie prowadzony dziennik samopoczucia. Aktywne użytkowanie aplikacji ma skłaniać do refleksji odnośnie stylu życia użytkownika oraz reagowania na zmiany zachodzące w samopoczuciu w efekcie analizy wygenerowanych statystyk. Aplikacja ma na celu propagowanie zdrowego stylu życia, bilansowania aktywności wykonywanych w trakcie dnia, dbania o odpowiednią dietę oraz higienę snu.

## 2. Opis funkcjonalności

Aplikację można użytkować na dwa sposoby: jako użytkownik zalogowany i niezalogowany. Użytkownik niezalogowany może skorzystać z następujących funkcjonalności:

- **Logowanie**

Logowanie jest możliwe poprzez uzupełnienie formularza logowania wpisując login oraz hasło. Po kliknięciu przycisku zaloguj następuje walidacja danych oraz jeśli przejdzie poprawnie, zmiana stanu użytkownika na użytkownika zalogowanego. Jeżeli nastąpi błąd walidacji, użytkownik zostanie o tym poinformowany.

- **Rejestracja**

Użytkownik ma możliwość zalogowania poprzez uzupełnienie formularza rejestracji wpisując swoje imię, nazwisko, email, login i hasło. Po zatwierdzeniu rejestracji następuje walidacja danych, jeżeli przejdzie poprawnie, użytkownik zmienia stan na użytkownika zalogowanego a jego dane zostają zapisane w bazie danych. Jeżeli nastąpi błąd walidacji, użytkownik zostanie o tym poinformowany.

Użytkownikowi zalogowanemu zostały udostępnione następujące funkcjonalności:

- **Dodawanie informacji o dniu dzisiejszym**

Użytkownik ma możliwość wprowadzenia informacji o swoim dniu poprzez odpowiedni formularz. W formularzu istnieje możliwość wprowadzenia następujących danych:

- a) Liczba przepracowanych godzin (< 24)
- b) Liczba spędzonych godzin na studiach (< 24)
- c) Liczba godzin spędzonych na nauce (< 24)
- d) Liczba przespanych godzin (< 24)
- e) Liczba godzin spędzonych na uprawianiu sportu (< 24)
- f) Liczba godzin spędzonych na odpoczynku (< 24)
- g) Liczba spożytych posiłków
- h) Ilość wypitej wody
- i) Liczba godzin spędzonych na spotkaniach towarzyskich (<24)
- j) Liczba spotkanych osób (>=0)
- k) Suplementy - nazwa i ilość zażytego suplementu (ilość suplementów jest nieograniczona jednak nazwa suplementu jest wymagana)
- l) Używki - nazwa i ilość zażytej używki (ilość używek jest nieograniczona jednak nazwa używki jest wymagana)

- m) Opis dnia
- n) Samopoczucie psychiczne (wymagane pole ze skalą od 1 do 10)
- o) Samopoczucie fizyczne (wymagane pole ze skalą od 1 do 10)

Po kliknięciu przycisku zapisz, dane są walidowane i jeśli wszystko jest poprawne, zostają zapisane. Jeśli nie, zostaje wysłany komunikat o błędzie.

- **Przeglądanie historycznych wpisów**

Użytkownik ma możliwość przeglądania skrótów historycznych wpisów (data oraz wartości samopoczucia fizycznego oraz psychicznego)

- **Edycja historycznego wpisu**

Użytkownik ma możliwość, po kliknięciu ikonki ołówka na historycznym wpisie, edycji każdego parametru w dokładnie taki sposób jak w przypadku dodawaniu nowego wpisu.

- **Usuwanie wpisu**

Użytkownik ma możliwość usunięcia historycznego wpisu poprzez kliknięcie ikony z koszem. Przed usunięciem wpisu, pokaże się alert z potwierdzeniem wyboru.

- **Wyświetlanie statystyk**

Użytkownik ma możliwość wyświetlania czterech rodzajów statystyk w określonym okresie czasu (ostatni tydzień, miesiąc, rok):

- a) **Zależność samopoczucia od wybranego parametru**

W tym przypadku wyświetlany jest zwykły wykres punktowy zależności parametru (oprócz samopoczucia fizycznego i psychicznego) od samopoczucia psychicznego lub fizycznego. Jeżeli więcej niż jeden dzień z danego przedziału czasowego ma dokładnie taką samą wartość parametru i samopoczucia, punkt staje się odpowiednio większy. Dodatkowo po najejchaniu myszką na dany punkt, zostaje wyświetlona etykieta z odpowiednimi wartościami x i y oraz krotnością występowania.

- b) **Zależność samopoczucia psychicznego od fizycznego**

Tym razem generowana jest heatmapa złożona z 10 na 10 pól odpowiednio dla każdej kombinacji wartości samopoczucia. W każdym polu wyświetlana jest krotność występowania danej kombinacji. Dodatkowo im większa krotność, tym ciemniejsze pole.

- c) **Zależność parametru od czasu**

Jest to wykres liniowy zależności wartości danego parametru w okresie wybranego czasu. Po najejchaniu myszką na obszar danego dnia, pojawia się etykieta z wartością parametru dla danego dnia.

- d) **Wartości statystyczne dla wybranego parametru**

Wyświetlany jest wykres kołowy przedstawiający procentowe rozmieszczenie wartości wybranego parametru. Dodatkowo podana jest wartość minimalna, maksymalna i średnia parametru w danym okresie czasu.

- **Wylogowanie**

Użytkownik ma możliwość wylogowania się, co sprawia, że jego stan zmienia się na użytkownika niezalogowanego.

### 3. Opis użytych technologii

Aplikacja składa się z części serwerowej, bazodanowej oraz klienckiej.

Część serwerowa została wykonana przy pomocy Spring Framework. Jest popularnym frameworkiem do tworzenia aplikacji Java. Umożliwia on szybkie i efektywne tworzenie aplikacji, zapewniając wiele gotowych rozwiązań i narzędzi. Dodatkowo został użyty mechanizm JPA (Java Persistence API). To specyfikacja ORM (Object-Relational Mapping), która umożliwia mapowanie obiektów Java na rekordy w bazie danych. JPA jest częścią standardu Java EE i dostarcza interfejsy do zarządzania relacyjnymi bazami danych.

Część kliencka została wykonana przy pomocy React.js. To biblioteka JavaScript do budowania interfejsów użytkownika. Jest oparty na komponentach, co umożliwia tworzenie modułowych i wielokrotnego użytku komponentów UI. React.js używa wirtualnego DOM (Virtual DOM), co przyspiesza manipulację i aktualizację elementów na stronie, co przekłada się na płynniejsze i wydajniejsze interakcje użytkownika.

W aplikacji została użyta relacyjna baza danych PostgreSQL a konkretnie ElephantSQL. To usługa hostingowa bazy danych w chmurze, która oferuje zarządzanie bazą danych PostgreSQL. Jest to popularny system zarządzania bazą danych relacyjnych, który zapewnia niezawodność, skalowalność i bogate funkcje. Wykorzystanie ElephantSQL pozwala na przechowywanie danych aplikacji w zdalnej bazie danych, z elastycznym skalowaniem zasobów w zależności od potrzeb.

W połączeniu, Spring z JPA, React.js i baza danych ElephantSQL tworzą kompleksowe rozwiązanie do tworzenia aplikacji internetowych. Spring z JPA dostarcza backendową logikę biznesową, obsługę persystencji danych i integrację z bazą danych. React.js pozwala na rozwijanie dynamicznych interfejsów użytkownika, które są szybkie, interaktywne i łatwe w utrzymaniu. Baza danych ElephantSQL zapewnia niezawodne przechowywanie danych i skalowalność systemu.

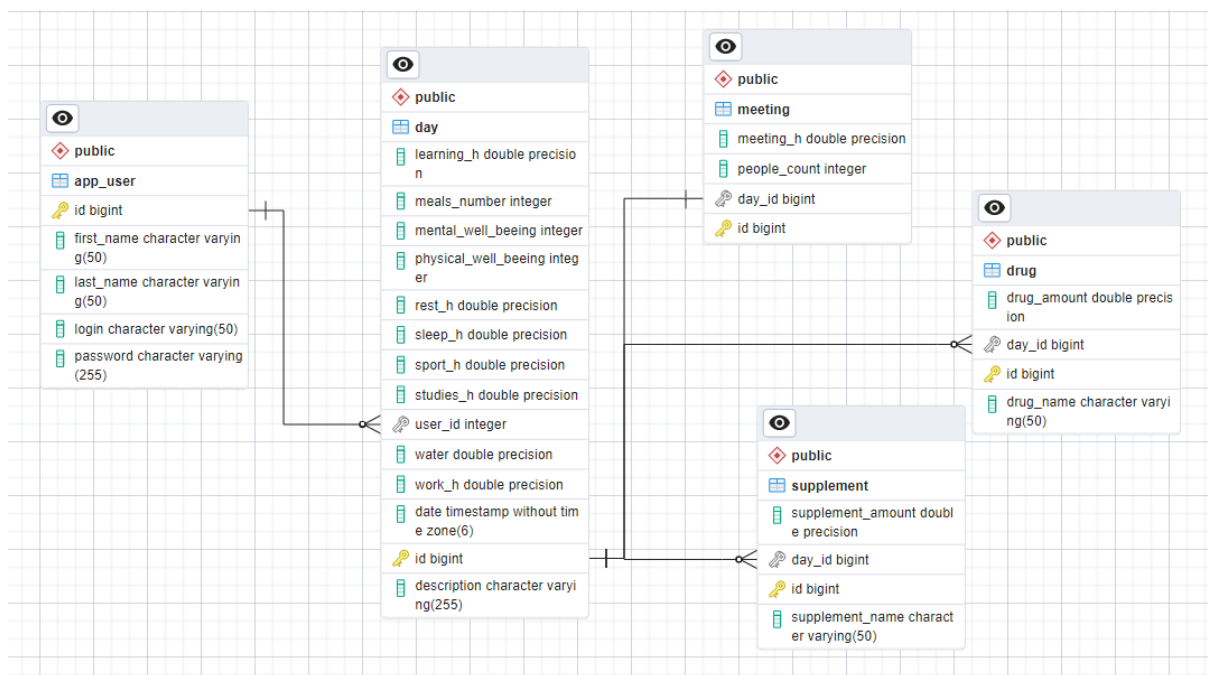
Autoryzacja w projekcie została zaimplementowana przy użyciu JWT (JSON Web Tokens).

Dokumentacja kodu części serwerowej została wykonana przy pomocy JavaDoc. Znajduje się w folderze Projekt\_Miszczak/projekt\_ZTI/target/site/apidoc

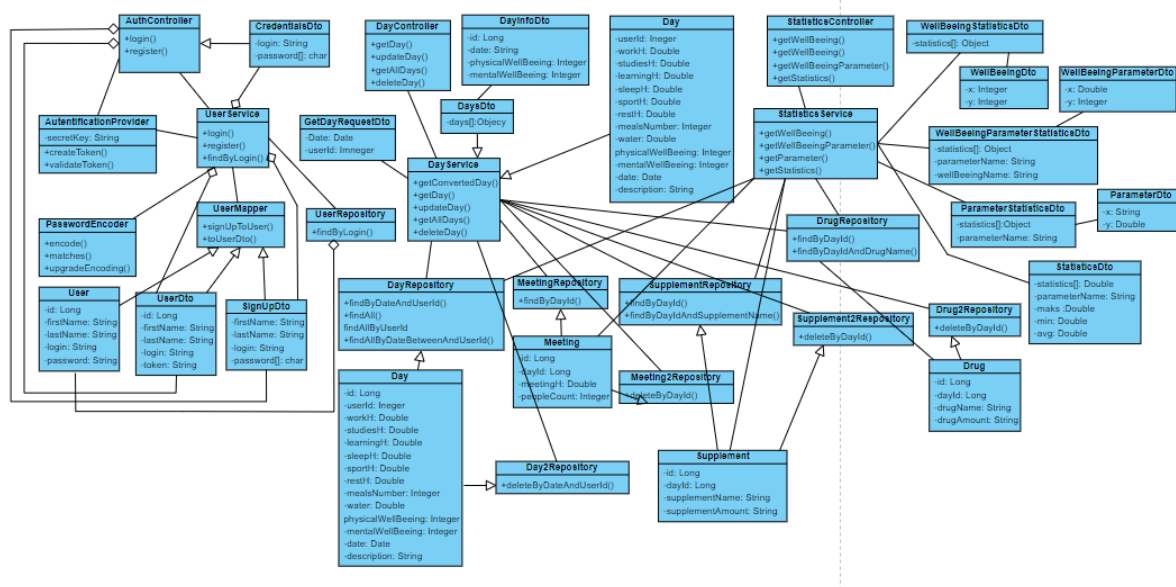
Dokumentacja kodu części klienckiej została wykonana przy pomocy JsDoc 3. Znajduje się w folderze Projekt\_Miszczak/client/out/

## 4. Diagramy UML

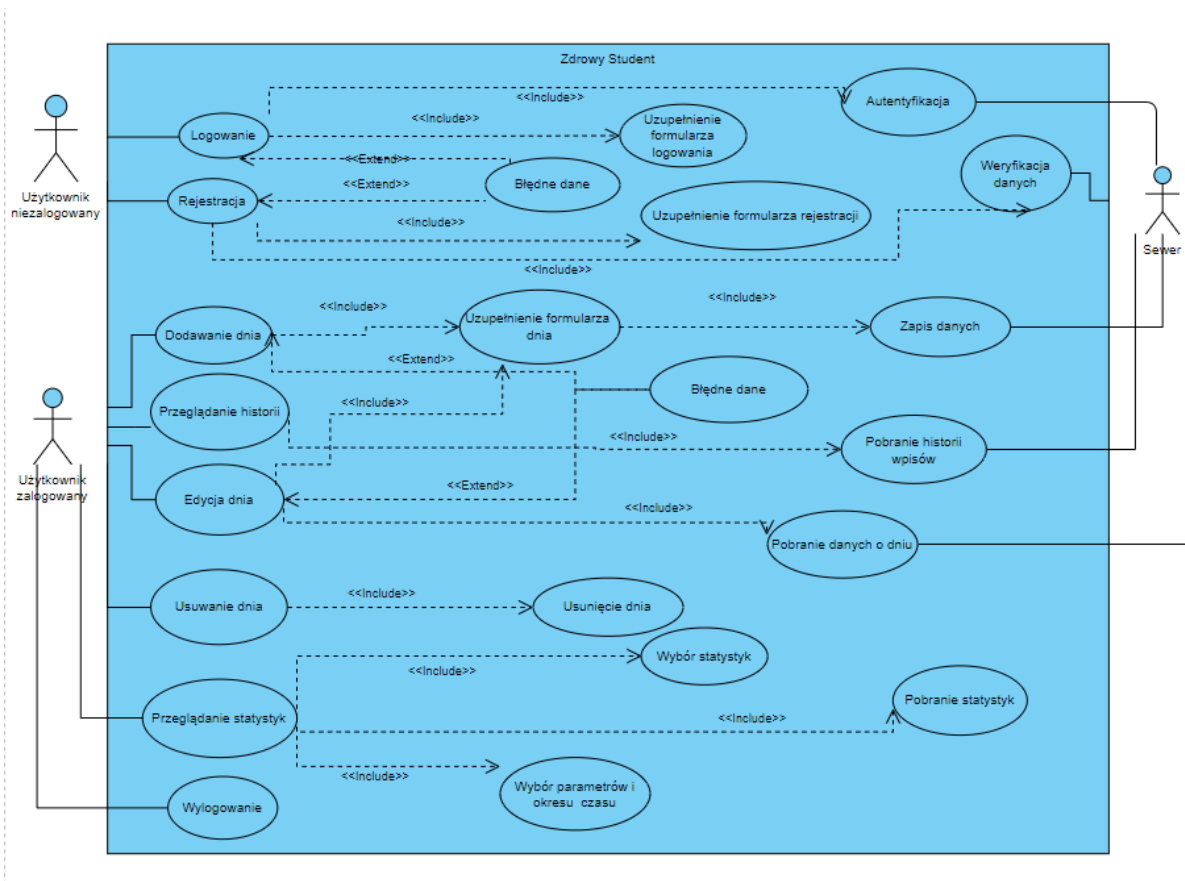
W ramach projektu zostały wykonane diagramy UML przedstawiające działanie aplikacji. Dodatkowo oprócz diagramów UML, został wygenerowany diagram ERD.



Rysunek 4. 1 Diagram ERD bazy danych



Rysunek 4.2 Diagram klas



Rysunek 4.3 Diagram UseCase

Powyższe diagramy są bardzo rozbudowane, co może utrudniać czytelność dlatego zostały umieszczone również w folderze głównym projektu.

## 5. Testy aplikacji

W części serwerowej zostały wykonane testy jednostkowe sprawdzające poprawność implementacji. Przetestowane zostały wszystkie metody kontrolerów oraz serwisów. Nie zostały wykonane testy repozytoriów, ponieważ korzystały one z JPA oraz CrudRepository. Są to części frameworka, więc jest on już dostatecznie dobrze przetestowany, dlatego można było założyć, że wszystko działa poprawnie. Testy zostały wykonane przy pomocy JUnit.

Szczegółowy opis testów jednostkowych został zamieszczony w dokumentacji kodu dołączona w folderze projektu (Projekt\_Miszcak/projekt\_ZTI/target/site/testapidoc). Wszystkie testy zostały uruchomione bez błędów.

Testowanie aplikacji klienckiej zostało sprowadzone do testów wdrożeniowych. Po dokeryzacji aplikacji i integracji z backendem, aplikacja została dogłębnie przetestowana. Obszary testowania aplikacji to:

- Zachowanie pożądane

Testowana była każda strona, podstrona i komponent, zapis, usuwanie i edycja danych, wysyłanie zapytań i przechwytywanie odpowiedzi z serwera. Testowane było również logowanie, wylogowanie, rejestracja oraz autoryzacja użytkownika. Wszystkie dane były wprowadzane prawidłowo (dane pożądane). Testowanie nie wskazało żadnych błędów.

- Błędne dane

Kolejnym krokiem było testowanie aplikacji na danych które były nieprawidłowe np. nieprawidłowy login w logowaniu, zbyt duża liczba godzin w formularzu dnia lub wpisanie nieistniejącego parametru (nazwy) w celu wygenerowania statystyk. Została przetestowana każda możliwa opcja błędnego wprowadzenia danych. W każdym przypadku została przekazana informacja o błędzie poprzez alert lub odpowiedni podpis. Świadczy to o poprawności działania aplikacji w przypadku występowania błędnych danych.

Opisane powyżej testy wskazały brak występowania błędów aplikacji oraz poprawność implementacji.

## 6. Instrukcja uruchomieniowa

Aby uruchomić aplikację należy wykonać następujące kroki:

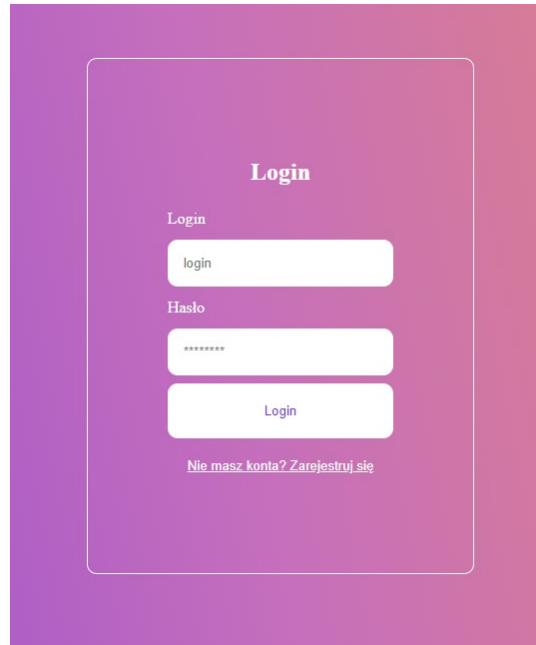
1. Upewnić się, że w środowisku uruchomieniowym zainstalowany jest Docker i Docker Compose.
2. Przejść do katalogu projektu Projekt\_Miszczak (katalog, w którym znajduje się plik docker-compose.yml).
3. Otworzyć terminal lub wiersz poleceń i wykonać następujące polecenie:  
`$ docker compose up`
4. Po zakończeniu procesu budowania i uruchamiania kontenerów, w terminalu powinny pojawić się logi aplikacji.
5. Serwer uruchomił się pod adresem <http://localhost:8088>  
Aplikacja klienta powinna uruchomić się pod adresem <http://localhost:80>
6. Aby zatrzymać działanie aplikacji, należy wykonać w terminalu polecenie:  
`$ docker compose down`

Aby uruchomić dokumentację części serwerowej, należy przejść do katalogu Projekt\_Miszczak/projekt\_ZTI/target/site/apidoc i włączyć plik index.html. Skutkuje to pokazaniem dokumentacji w przeglądarce.

Aby uruchomić dokumentację części klienckiej należy zrobić analogicznie jak w przypadku części serwerowej, jednak plik index.html znajduje się w katalogu Projekt\_Miszczak/client/out/

## 7. Podstawowy podręcznik użytkownika

Po uruchomieniu aplikacji klienckiej, na ekranie pojawia się formularz logowania:



The screenshot shows a login form titled "Login" centered on a purple-to-pink gradient background. The form is enclosed in a white rounded rectangle. It contains two input fields: "Login" with the text "login" and "Hasło" with masked characters "\*\*\*\*\*". Below these is a "Login" button. At the bottom, there is a link that says "Nie masz konta? Zarejestruj się".

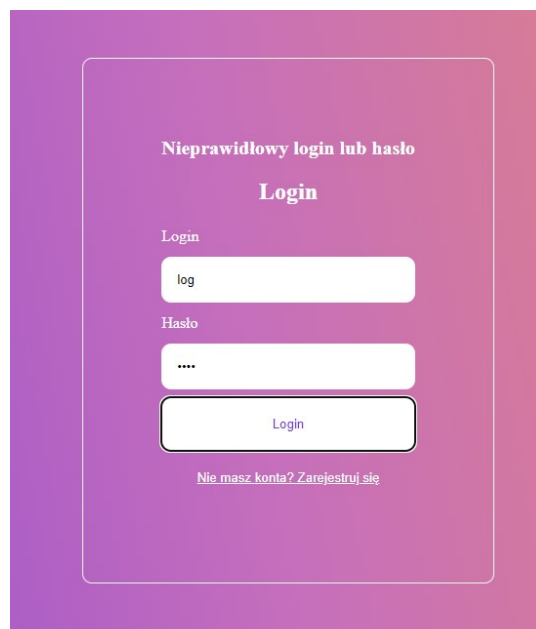
Rysunek 7.1 Formularz logowania

Dane testowego użytkownika:

Login: login

Hasło: pass

Po niepoprawnym wpisaniu danych logowania, otrzymujemy komunikat o błędzie:



The screenshot shows the same login form as before, but with an error message "Nieprawidłowy login lub hasło" displayed above the "Login" title. The input fields now contain "log" and "\*\*\*\*". The "Login" button and the registration link remain at the bottom.



### Rysunek 7.2 Komunikat o błędzie logowania

Aby się zarejestrować, należy kliknąć biały napis i pojawi się formularz logowania:

The image shows a registration form titled "Rejestracja" (Registration) centered on a purple-to-pink gradient background. The form is enclosed in a white rounded rectangle. It contains the following elements: a title "Rejestracja", four input fields labeled "Imię" (Name), "Nazwisko" (Surname), "Login", and "Hasło" (Password), a "Zarejestruj się" (Register) button, and a link "Masz już konto? Zaloguj się" (Already have an account? Log in).

### Rysunek 7.3 Formularz logowania

Po wpisaniu danych należy kliknąć przycisk „Zarejestruj się”. Jeżeli podany login już istnieje, zostanie wyświetlony komunikat o błędzie:

The image shows the same registration form as in Figure 7.2, but with a message at the top: "Użytkownik o podanym loginie już istnieje" (User with the given login already exists). The form is titled "Rejestracja". The input fields are now filled with the following text: "Imię" (Name) is "Jan", "Nazwisko" (Surname) is "Kowalski", "Login" is "login", and "Hasło" (Password) is "\*\*\*\*". The "Zarejestruj się" button and the link "Masz już konto? Zaloguj się" are still present at the bottom.

Rysunek 7.4 Komunikat o błędzie przy rejestracji

Jeżeli chcemy przejść do formularza logowania, należy kliknąć biały napis. Zarówno w przypadku logowania jak i rejestracji, jeśli dane są poprawne, po kliknięciu przycisku potwierdzającego, użytkownik zostaje zalogowany i przeniesiony do formularza aktualnego dnia.

The screenshot shows a web application interface with a purple header bar. On the right side of the header, there are links: 'DZIŚ', 'HISTORIA', 'STATYSTYKI', and 'WYLOGUJ SIĘ'. The main content area has a light blue background. In the center, there is a white box titled 'Opisz swój dzień'. Inside this box, there is a list of ten items, each with a label and a value: 'Liczba przemieszczonych godzin' (1), 'Liczba godzin spędzonych na studiach', 'Liczba godzin spędzonych na zajęciach', 'Liczba przemieszczonych godzin', 'Liczba godzin spędzonych na wykładach specjalizacji', 'Liczba godzin spędzonych na wykładach specjalizacji', 'Liczba godzin spędzonych na wykładach specjalizacji', 'Liczba godzin spędzonych na wykładach specjalizacji', 'Liczba godzin spędzonych na wykładach specjalizacji', and 'Liczba godzin spędzonych na wykładach specjalizacji'. Below this list, there is a section for 'Dodaj suplement' and 'Dodaj używkę'. There is a text input field for 'Wprowadź suplement...'. Below that, there are two sections for 'Samopoczucie psychiczne' and 'Samopoczucie fizyczne', each with a dropdown menu showing '5' and '7' respectively. At the bottom of the form, there is a 'Zapisz' button.

Rysunek 7.5 Ogólny wygląd aplikacji

## Opisz swój dzień

Liczba przepracowanych godzin

8

Liczba godzin spędzonych na studiach

2

Liczba godzin spędzonych na nauce

2

Liczba przespanych godzin

6

Liczba godzin spędzonych na uprawianiu sportu

2

Liczba godzin poświęconych na odpoczynek

2

Liczba spożytych posiłków

4

Ilość wypitej wody (w litrach)

2

Ilość godzin na spotkaniach towarzyskich

2

Liczba spotkanych osób

1

Dodaj suplement

Nazwa suplementu

Witamina C

Ilość

100



Dodaj używkę

Nazwa używki

Kawa

Ilość

2



Notatka

Samopoczucie psychiczne

9 ▼

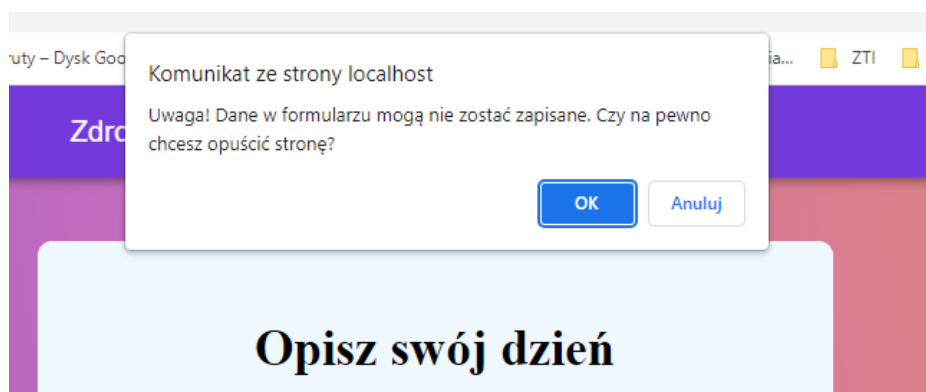
Samopoczucie fizyczne

10 ▼

Zapisz

Rysunek 7.6 Formularz dnia

Użytkownik ma możliwość uzupełnienia pól. Jeżeli uzupełni formularz, nie zapisując go i będzie chciał przejść do innej zakładki poprzez kliknięcie np. na napis „Historia” w nagłówku, pojawi się alert potwierdzający.



Rysunek 7.7 Alert braku zapisania formularza.

Po kliknięciu „OK”, użytkownik zostaje przeniesiony w oczekiwane miejsce bez zapisania formularza. Jeżeli kliknie „Anuluj”, pozostanie na aktualnej stronie.

Zapisywanie wartości odbywa się poprzez kliknięcie przycisku z podpisem „Zapisz”. Aby dodać suplement lub używkę, należy kliknąć przycisk z odpowiednim podpisem. Żeby usunąć suplement lub używkę, należy kliknąć czerwony krzyżyk obok usuwanego obiektu.

Jeżeli, po kliknięciu przycisku zapisującego dane, dane okażą się niepoprawne, użytkownik zostanie o tym poinformowany:

## Opisz swój dzień

Liczba przepracowanych godzin

56

Wartość pola musi być z przedziału od 0 do 24

Liczba godzin spędzonych na studiach

2

Liczba godzin spędzonych na nauce

2

Liczba przespanych godzin

6

Liczba godzin spędzonych na uprawianiu sportu

2

Liczba godzin poświęconych na odpoczynek

2

Liczba spożytych posiłków

4

Ilość wypitej wody (w litrach)

2

Ilość godzin na spotkaniach towarzyskich

2

Liczba spotkanych osób

1

Dodaj suplement

Nazwa suplementu	Ilość
	100

✗ Musisz uzupełnić nazwę

Dodaj używkę

Nazwa używki	Ilość
Kawa	2

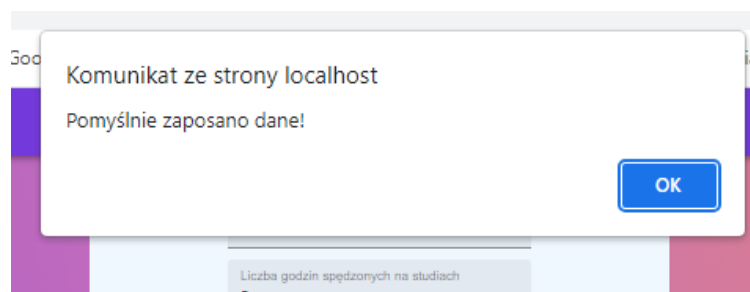
✗

Notatka

Samonadzór nieuczciwy

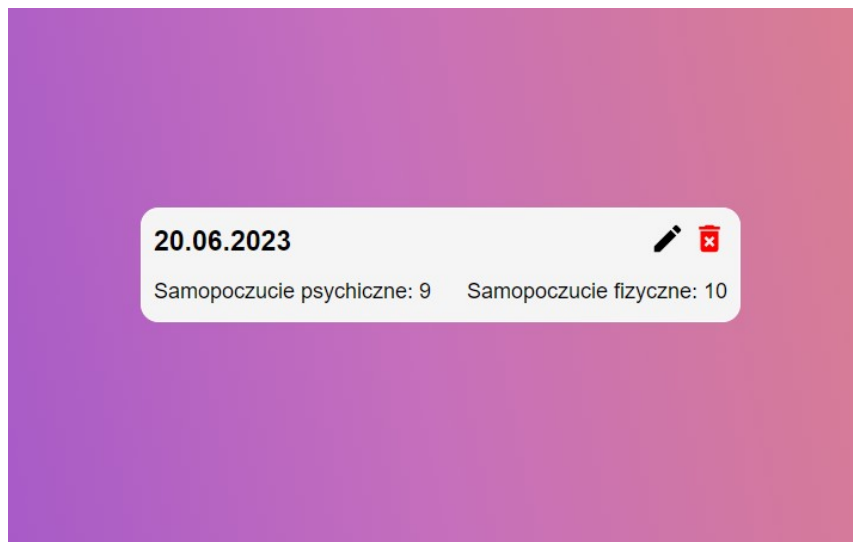
Rysunek 7. 8 Komunikaty o błędach

Jeżeli dane są prawidłowe, zostanie wyświetlony alert o pomyślnym wykonaniu operacji



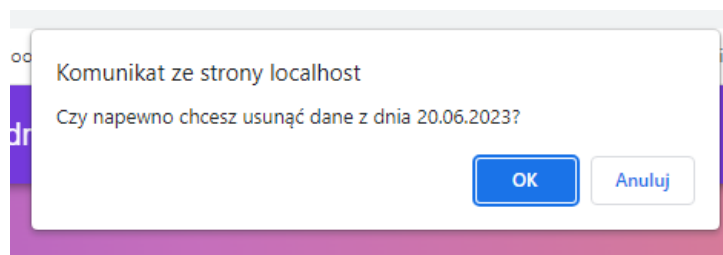
Rysunek 7.9 Komunikat o poprawnym zapisie

Jeżeli klikniemy na zakładkę „Historia”, zostanie pokazana historia wszystkich zapisanych dni:



Rysunek 7.8 Historia zapisanych dni

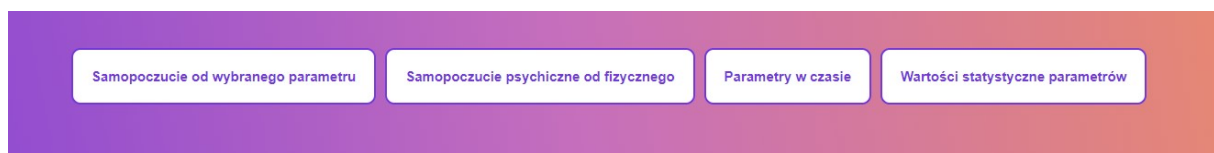
Żeby edytować historyczny wpis, należy kliknąć ołówek, zostaniemy wtedy przeniesieni do dokładnie takiego samego formularza jak na Rysunku 7.6 jednak jego pola są uzupełnione zgodnie z wcześniejszym wpisem. Zasada jego działania jest dokładnie taka sama jak w przypadku dodawania dnia. Aby usunąć wpis, należy kliknąć czerwony kosz na śmieci. Zostanie wtedy wyświetlony alert:



Rysunek 7.9 Alert potwierdzający usunięcie dnia

Jeżeli klikniemy „OK”, wpis zostanie usunięty, w przeciwnym razie nic się nie stanie.

Jeżeli klikniemy na pole „Statystyki” w nagłówku, zostaniemy przeniesieni do strony ze statystykami.:



Rysunek 7.10 Defaultowa strona ze statystykami

Jeżeli klikniemy na któryś z przycisków, pokażą się możliwe parametry do wyboru:



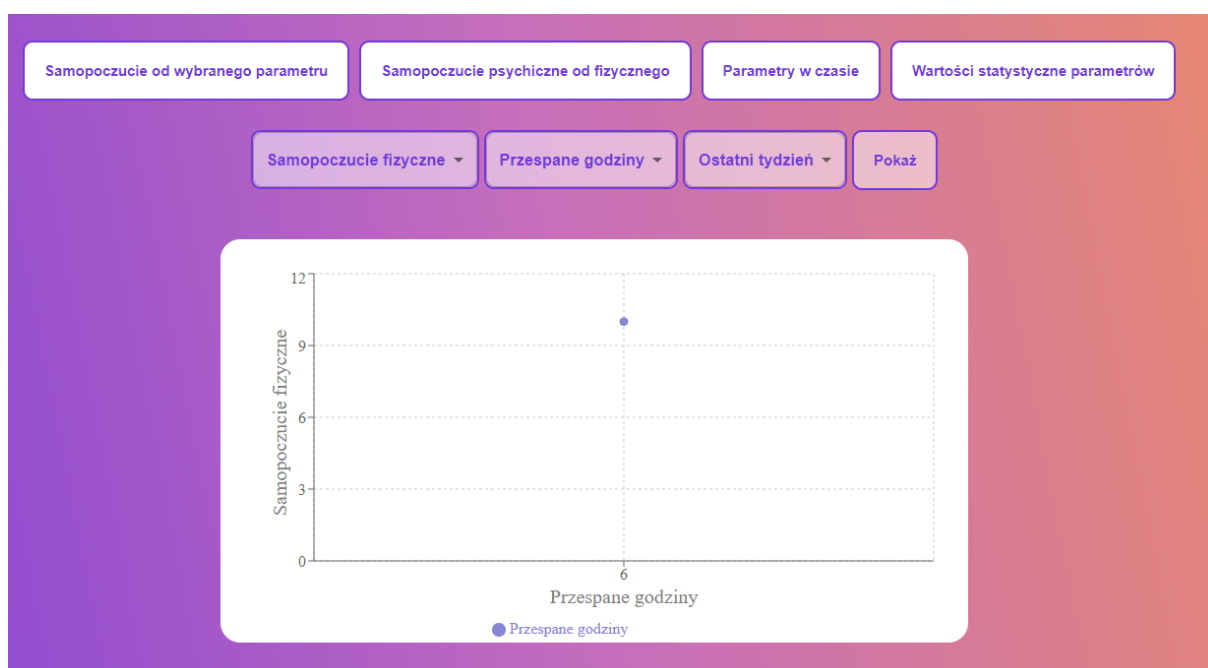
Rysunek 7.11 Opcje wyboru parametrów dla pierwszego przycisku

Jeżeli chcemy wybrać używkę lub suplement, należy wpisać jej nazwę:



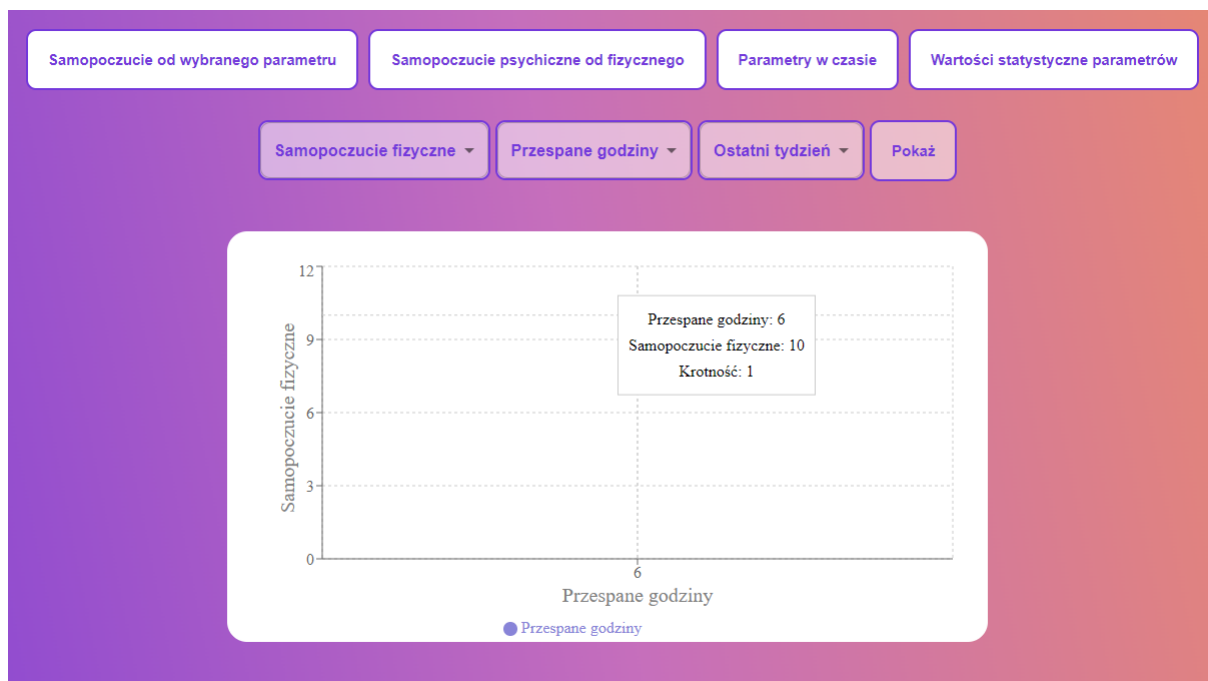
Rysunek 7.12 Wybór używki

Po kliknięciu przycisku „Pokaż”, pokazuje się wybrana statystyka:



Rysunek 7.13 Wybrany wykres statystyczny

Zasada działania przy każdym innym rodzaju statystyk jest dokładnie taka sama. W przypadku wykresów „Samopoczucie od wybranego parametru” oraz „Parametry w czasie”, po najechnaniu myszką na okolice punktu, pojawia się etykieta z dodatkowymi informacjami:



Rysunek 7.14 Wykres z pokazaną etykietą

Jeżeli użytkownik kliknie na „Wyloguj się” w nagłówku, zostaje wylogowany i przeniesiony do formularza logowania.

## 8. Podsumowanie

Wykonany projekt działa poprawnie i realizuje stawiane wymagania. Wykorzystane technologie bardzo usprawiły proces tworzenia aplikacji. Aplikacja jest w pełni funkcjonalna i potrzebna. Istnieje wiele możliwości długotrwałego rozwoju projektu takie jak dodanie większej ilości statystyk, rozbudowanie funkcjonalności notatek poprzez dodawanie np. zdjęć lub obrazów oraz wprowadzenie planów i alertów bezpieczeństwa z propozycją rozwiązania problemu w przypadku długotrwałego złego samopoczucia.