

# Dokumentacja aplikacji Hoacc

## 1. WYMAGANIA BIZNESOWE

Aplikacja webowa o nazwie „Hoacc” ma za zadanie wspierać zarządzanie budżetem domowym poprzez:

- monitorowanie wydatków i przychodów: umożliwienie rejestrowania oraz kategoryzowania wydatków i przychodów użytkownika, oferując prosty interfejs do dodawania oraz edycji transakcji,
- analizę i wykresy: prezentację analizy finansów w formie wykresów, umożliwiając szybką ocenę i zrozumienie aktualnej sytuacji finansowej,
- przegląd operacji: dostęp do przejrzystej historii transakcji,
- zarządzanie powiadomieniami: konfigurację powiadomień o ważnych transakcjach, zbliżających się terminach płatności,
- cele finansowe: możliwość ustalania i monitorowania celów finansowych (np. oszczędności, wydatki na konkretne cele) oraz śledzenie postępów w ich realizacji,
- bezpieczeństwo: zapewnienie bezpiecznego przechowywania danych finansowych,
- wsparcie multiplatformowe: dostępność aplikacji na różnych urządzeniach z zachowaniem spójności danych między nimi,
- elastyczność i łatwość użytkowania: stworzenie intuicyjnego interfejsu, zapewniającego łatwą nawigację dla użytkowników o różnym poziomie znajomości technologii.

Dodatkowo, język angielski został wybrany jako język interfejsu użytkownika, umożliwiający obsługę aplikacji przez użytkowników na całym świecie. Te funkcjonalności mają umożliwić kompleksowe wsparcie w zarządzaniu budżetem domowym, zapewniając użytkownikowi skuteczne narzędzia do monitorowania, analizy i zarządzania własnymi finansami.

## 2. WYMAGANIA FUNKCJONALNE UŻYTKOWNIKA

Wymagania funkcjonalne użytkownika, które powinna spełniać aplikacja to:

- rejestracja konta użytkownika: umożliwienie użytkownikowi rejestracji poprzez podanie podstawowych informacji, takich jak adres e-mail i hasło. Weryfikacja unikalności adresu e-mail podczas rejestracji,
- przechowywanie hasła: zapewnienie bezpiecznego przechowywania danych użytkownika, takich jak hasło za pomocą funkcji BCrypt,
- logowanie użytkownika: udostępnienie formularza logowania, gdzie użytkownik może wprowadzić dane uwierzytelniające (e-mail i hasło) w celu dostępu do konta,
- zarządzanie profilem użytkownika: możliwość zmiany hasła,
- obsługa błędów: wyświetlanie komunikatów o błędach lub ostrzeżeń użytkownikowi w przypadku niepowodzenia wykonania operacji podczas korzystania z aplikacji. Zapewnienie odpowiednich komunikatów dla użytkownika w przypadku problemów z funkcjonalnościami aplikacji (np. awarie serwera),
- wylogowywanie: udostępnienie opcji wylogowania się z konta, zapewniającej zabezpieczenie w razie dostępu do aplikacji z publicznego urządzenia lub komputera,
- kategorie: każdy użytkownik systemu ma dostęp do domyślnych kategorii. Użytkownik nie ma możliwości zmiany kategorii,
- operacje: każdy użytkownik systemu może dodać nową operację. Dane które użytkownik musi zapewnić to :
  - wybór typu operacji spośród Przychód/Wydatek,
  - wybór z dostępnych kategorii spośród Inne/Rachunki/Żywność/Edukacja/Rozrywka,
  - data operacji,
  - opis,
  - kwota.

W formularzu dane są sprawdzane pod kątem poprawności. Kwota musi być liczbą o maksymalnie 10 cyfrach, z dokładnością do dwóch miejsc po przecinku, a opis może zawierać maksymalnie 50 znaków,

- zarządzanie operacjami: każdy użytkownik systemu może zarządzać wcześniej dodanymi operacjami w zakresie:
  - zmiany typu operacji,
  - zmiany kategorii,
  - zmiany daty operacji,
  - zmiany opisu,

- zmiany kwoty,
- usunięcie operacji.

- każdy użytkownik systemu ma możliwość przeglądania historii swoich operacji,
- przypomnienia: każdy użytkownik systemu może dodać nowe przypomnienie. Dane które użytkownik musi zapewnić to :
  - tytuł,
  - kwota,
  - data przypomnienia.

W formularzu dane są sprawdzane pod kątem poprawności. Kwota musi być liczbą o maksymalnie 10 cyfrach, z dokładnością do dwóch miejsc po przecinku, a tytuł może zawierać maksymalnie 20 znaków,

- zarządzanie powiadomieniami: każdy użytkownik może usunąć wcześniej dodane przez siebie powiadomienie,
- cele: każdy użytkownik systemu może dodać nowe cele. Dane które użytkownik musi zapewnić to :
  - nazwa,
  - aktualna kwota,
  - docelowa kwota.

W formularzu dane są sprawdzane pod kątem poprawności. Kwota musi być liczbą o maksymalnie 10 cyfrach, z dokładnością do dwóch miejsc po przecinku, a nazwa może zawierać maksymalnie 20 znaków,

- zarządzanie celami: każdy użytkownik może zarządzać wcześniej dodanymi celami w zakresie:
  - zmiany nazwy,
  - zmiany aktualnej kwoty,
  - zmiany docelowej kwoty,
  - usunięcia celu.
- każdy użytkownik systemu może przeglądać wizualizację sumy przychodów i sumy wydatków na wykresie słupkowym dla wyszczególnionych okresów czasu. Możliwy wybór okresów czasu to:
  - dzisiaj,
  - wczoraj,
  - ostatnie 7 dni,

- ostatnie 30 dni,
  - ostatnie 90 dni,
  - ostatnie 6 miesięcy,
  - ostatni rok.
- każdy użytkownik systemu może przeglądać wizualizację różnicy przychodów i wydatków na wykresie kolumnowym dla wyszczególnionych okresów czasu. Możliwy wybór okresów czasu to:
    - ostatnie 3 miesiące,
    - ostatnie 6 miesięcy,
    - ostatni rok.
- każdy użytkownik systemu może przeglądać wizualizację sumy wydatków dla poszczególnych kategorii na wykresie kołowym dla wyszczególnionych okresów czasu. Możliwy wybór okresów czasu to:
    - dzisiaj,
    - wczoraj,
    - ostatnie 7 dni,
    - ostatnie 30 dni,
    - ostatnie 90 dni,
    - ostatnie 6 miesięcy,
    - ostatni rok.

Te wymagania funkcjonalne stanowią podstawę dla aplikacji webowej umożliwiającej użytkownikom zarządzanie budżetem domowym.

### 3. WYMAGANIA NIEFUNKCJONALNE UŻYTKOWNIKA

Wymagania niefunkcjonalne użytkownika, które powinna spełniać aplikacja to:

- intuicyjny interfejs użytkownika: zapewnienie prostego i łatwego w obsłudze interfejsu, umożliwiającego intuicyjne nawigowanie po aplikacji,
- szybkość i wydajność: minimalizacja czasu ładowania się stron i szybka reakcja na interakcje użytkownika, aby zapewnić płynne doświadczenie,

- dostępność na różnych urządzeniach: kompatybilność aplikacji z różnymi urządzeniami, takimi jak smartfony, tablety i komputery, aby umożliwić dostęp użytkownikom z różnych platform.

## 4. WYMAGANIA NIEFUNKCJONALNE SYSTEMOWE

Wymagania niefunkcjonalne systemowe, które powinno spełniać stworzone oprogramowanie służące do efektywniejszego zarządzania budżetem to:

- kompatybilność przeglądarek: aplikacja powinna być kompatybilna z różnymi przeglądarkami internetowymi, zapewniając spójność działania niezależnie od wybranej przez użytkownika przeglądarki,
- skalowalność: system powinien być łatwo skalowalny, umożliwiając elastyczne dostosowanie do wzrostu liczby użytkowników i potrzeb infrastrukturalnych,
- bezpieczeństwo: zapewnienie wysokiego poziomu ochrony danych poprzez zastosowanie odpowiednich mechanizmów szyfrowania, autoryzacji i uwierzytelniania użytkowników.

## 5. WYKORZYSTANE TECHNOLOGIE

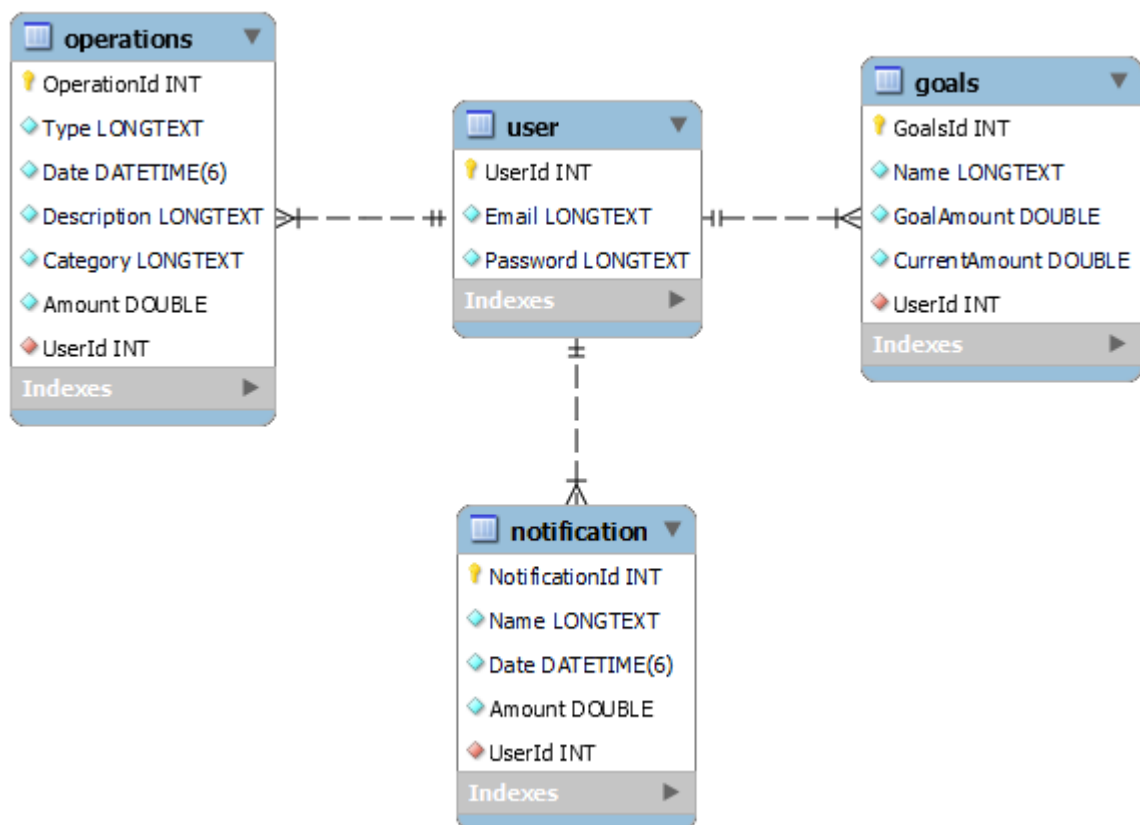
Technologie wykorzystane podczas tworzenia aplikacji Hoacc :

- Vue.js
- TypeScript
- Tailwind CSS
- Apex Charts
- Chart.js
- ASP.NET Core
- Entity Framework Core
- MySQL

## 6. DIAGRAM ENCJI

W poniższym podrozdziale przedstawiono graficzną reprezentację struktury danych i ich wzajemnych zależności w kontekście omawianej aplikacji webowej, które zilustrowane są na rys.

1. Diagram encji jest kluczowym narzędziem, które umożliwia zobrazowanie relacji między poszczególnymi elementami danych oraz ich organizację. Poprzez jego analizę lepiej rozumiemy strukturę informacji oraz relacje między nimi, co jest kluczowe dla pełniejszego zrozumienia funkcjonowania omawianej aplikacji.



Rys.1. Diagram encji

Użytkownik:

- użytkownik jest identyfikowany za pomocą UserId,
- użytkownik posiada unikalny adres email,
- hasło użytkownika przechowywane jest jako ciąg zaszyfrowanych znaków,
- każdy użytkownik ma możliwość posiadania wielu: operacji, przypomnień, celów.

Operacje:

- operacja jest identyfikowana za pomocą OperationId,
- operacja jest powiązana z użytkownikiem i nie może istnieć bez niego,

- każda operacja musi posiadać typ operacji,
- każda operacja musi mieć określoną datę,
- każda operacja musi posiadać opis,
- każda operacja musi należeć do określonej kategorii,
- każda operacja musi mieć zadeklarowaną kwotę.

Cele:

- cel jest identyfikowany za pomocą GoalsId,
- cel jest powiązany z użytkownikiem i nie może istnieć bez niego,
- każdy cel musi posiadać nazwę,
- każdy cel musi mieć zadeklarowaną kwotę docelową,
- każdy cel musi mieć zadeklarowaną kwotę aktualną.

Powiadomienia:

- powiadomienie jest identyfikowane za pomocą NotificationId,
- powiadomienie jest powiązane z użytkownikiem i nie może istnieć bez niego,
- każde powiadomienie musi posiadać nazwę,
- każde powiadomienie musi mieć określoną datę,
- każde powiadomienie musi mieć zadeklarowaną kwotę.

## 7. ADRESOWANIE W APLIKACJI HOACC

Aplikacja webowa do komunikacji między klientem a serwerem korzysta z takich metod adresowania jak metody HTTP: GET, POST i DELETE. Każda z tych metod pełni określoną rolę w przekazywaniu żądań oraz manipulowaniu danymi w omawianej aplikacji.

Metoda GET służy do pobierania danych z określonego zasobu, zazwyczaj poprzez żądanie danych od serwera. W aplikacji wykorzystano tę metodę do pobierania informacji, takich jak operacje danego użytkownika, powiadomienia, cele czy dane użytkownika.

Metoda POST umożliwia przesyłanie danych do serwera, często wykorzystywana do wysyłania formularzy, dodawania nowych zasobów do bazy danych czy też dokonywania zmian w istniejących danych. Omawiana aplikacja wykorzystuje tę metodę do przekazywania danych z formularzy logowania, rejestracji czy też do dodawania nowych operacji, celów lub powiadomień.

Metoda DELETE służy do usuwania określonych zasobów na serwerze. W aplikacji jest wykorzystywana do usuwania operacji, celów czy chociażby powiadomień, które użytkownik zdecyduje się usunąć.

W omawianej aplikacji aby zarządzanie i identyfikacja zasobów były łatwiejsze zastosowano identyczne prefiksy dla powiązanych ze sobą stron. Dla operacji dotyczących danych z operacjami użytkownika zastosowano prefiks /Operations. Dla funkcji serwisowych dotyczących użytkownika użyto prefiksu /User. Natomiast dla operacji dotyczących celów i powiadomień użyto prefiksu /Goals oraz /Notification. Jeżeli chodzi o uwierzytelnianie skorzystano z prefiksu /auth/login.

## 8. ZABEZPIECZENIE ADRESÓW PO STRONIE SERWERA

W celu zapewnienia bezpieczeństwa dostępu do zasobów po stronie serwera, wykorzystano tokeny JSON Web Token (JWT). Jest to sposób autoryzacji, który umożliwia uwierzytelnienie użytkownika i kontrolę dostępu do określonych adresów w aplikacji.

Dodatkowo aby zwiększyć bezpieczeństwo ustawiany jest czas wygaśnięcia tokenu na jedną godzinę, po jego wygaśnięciu użytkownik musi zalogować się ponownie.

W przypadku gdy użytkownik systemu próbuje uzyskać dostęp do określonych adresów, token JWT jest przesyłany w nagłówku zapytania HTTP, następnie jest on weryfikowany pod kątem autentyczności i jeżeli jest poprawny przyznaje dostęp użytkownikowi do żądanych zasobów.

## 9. ZABEZPIECZENIE ADRESÓW PO STRONIE KLIENTA

Bezpieczeństwo po stronie klienta jest kluczowe dla ochrony danych użytkownika oraz zapewnienia bezpiecznej interakcji z aplikacją. W omawianej aplikacji webowej, w przypadku zabezpieczeń adresów po stronie klienta, walidacja danych wejściowych koncentruje się na bezpiecznym przyjmowaniu oraz przetwarzaniu informacji wprowadzanych przez użytkownika i obejmuje ona m.in. sprawdzenie poprawności formatu danych, zabezpieczenia przed błędnymi danymi czy komunikacje błędów.

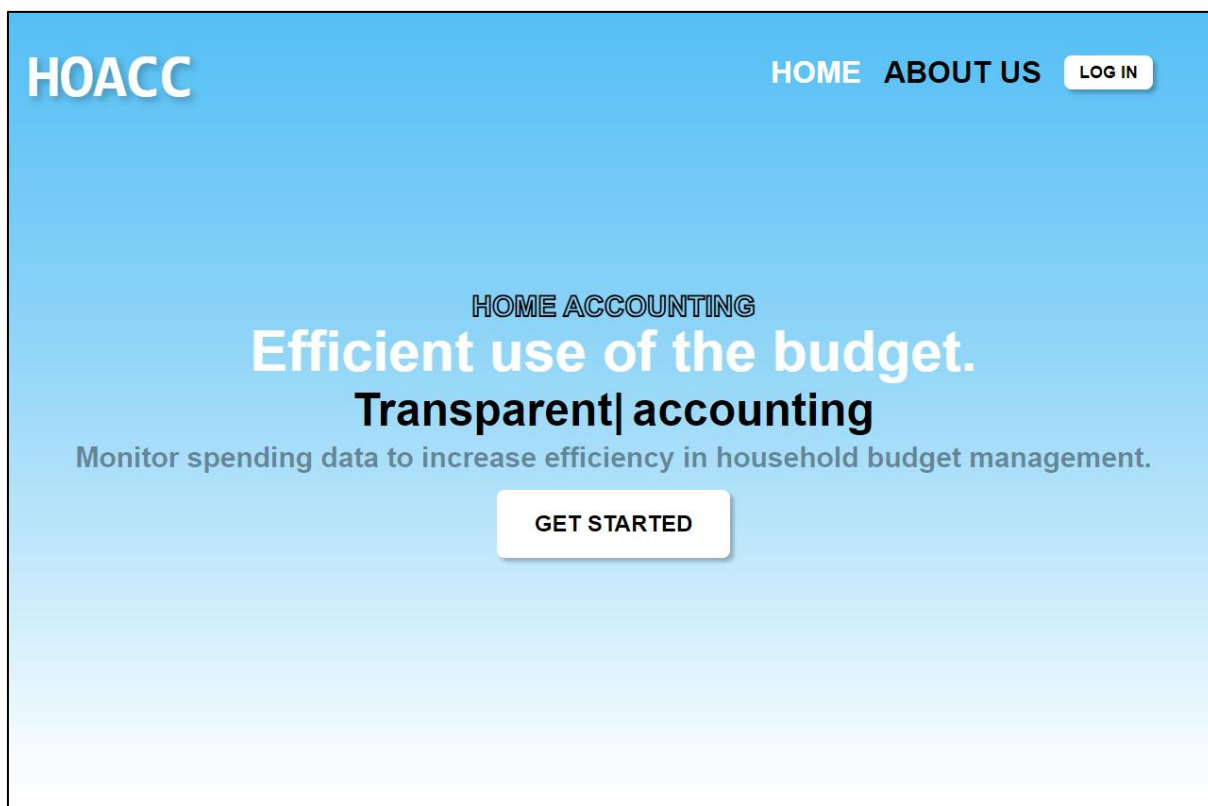
Aplikacja Hoacc aby zwiększyć bezpieczeństwo po stronie klienta korzysta również z zaszyfrowanych połączeń między klientem a serwerem, które są osiągane z pomocą protokołu komunikacji HTTPS (ang. – Hypertext Transfer Protocol Secure, pol.- bezpieczny protokół przesyłania



hipertekstu), który ma znaczny wpływ na bezpieczną transmisję danych. Ponad to omawiana aplikacja autoryzuje token użytkownika. Dzięki temu mechanizmowi aplikacja może kontrolować, jakie działania i zasoby są dostępne dla danego użytkownika, ograniczając tym samym ryzyko nieautoryzowanego dostępu do danych. Autoryzacja tokenu użytkownika pozwala także na wygaśnięcie i odświeżanie tokenów, co dodatkowo zwiększa bezpieczeństwo danych przechowywanych i przesyłanych po stronie klienta.

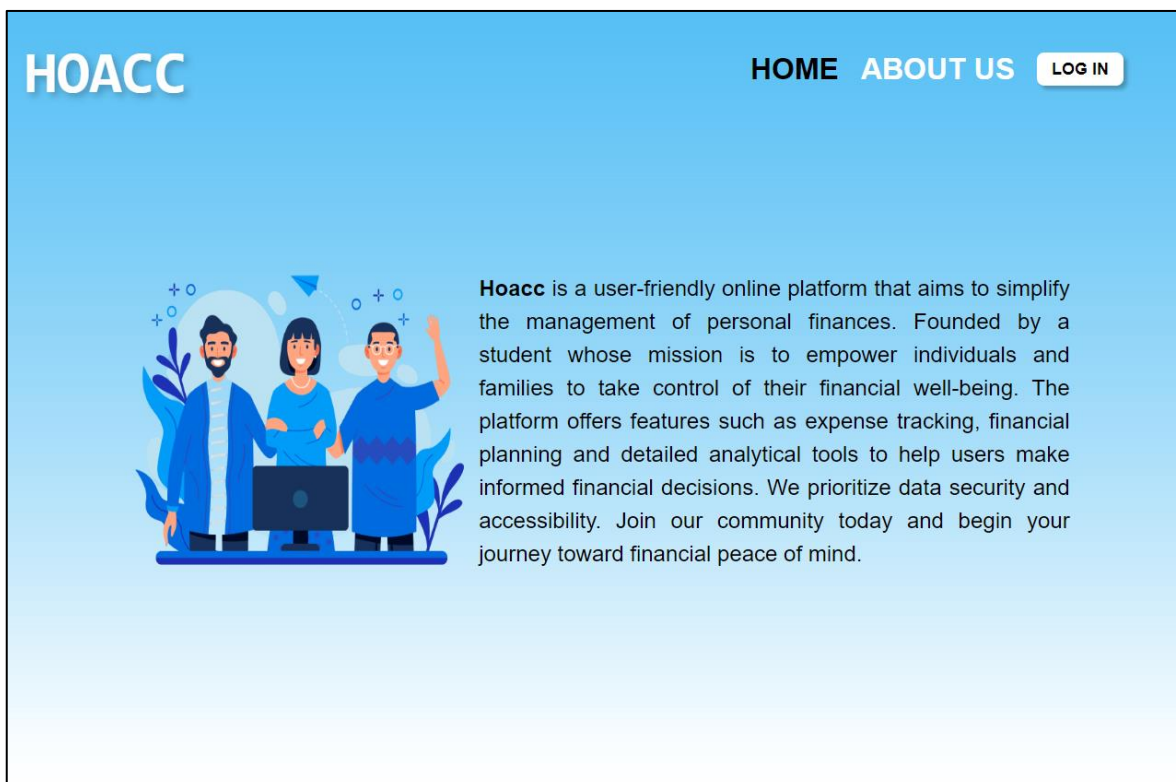
## 10. INTERFEJS GRAFICZNY

Rys. 5.8. ukazuje stronę startową aplikacji, która jest widoczna po wejściu na aplikację webową.



Rys. 5.8. Widok ekranu startowego aplikacji

Rys. 5.10 przedstawia stronę z informacjami o aplikacji, takimi jak ogólne założenia i jej główne funkcjonalności, dostępna jest po wciśnięciu przycisku „ABOUT US” znajdującego się na panelu nawigacyjnym w prawym górnym rogu aplikacji.



Rys. 5.10. Widok ekranu zawierającego informacje na temat aplikacji

Na rys. 5.13 oraz rys. 5.14 przedstawione są formularze logowania oraz rejestracji, do których dostęp uzyskujemy naciskając kolejno przycisk „LOG IN” lub „GET STARTED” na stronie startowej lub bezpośrednio naciskając napis „Login” w formularzu rejestracji lub „Sign up” w formularzu logowania.

**HOACC**

**Login**

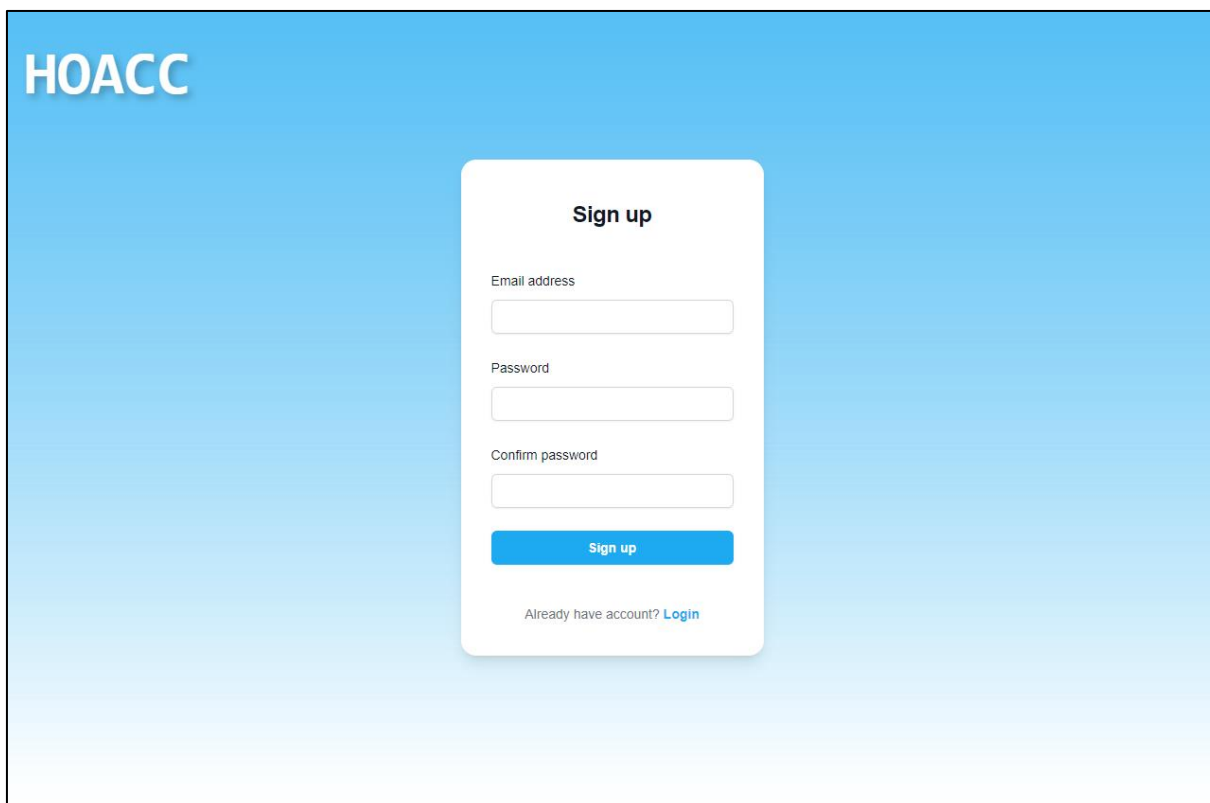
Email address

Password

Login

Not a member? [Sign up](#)

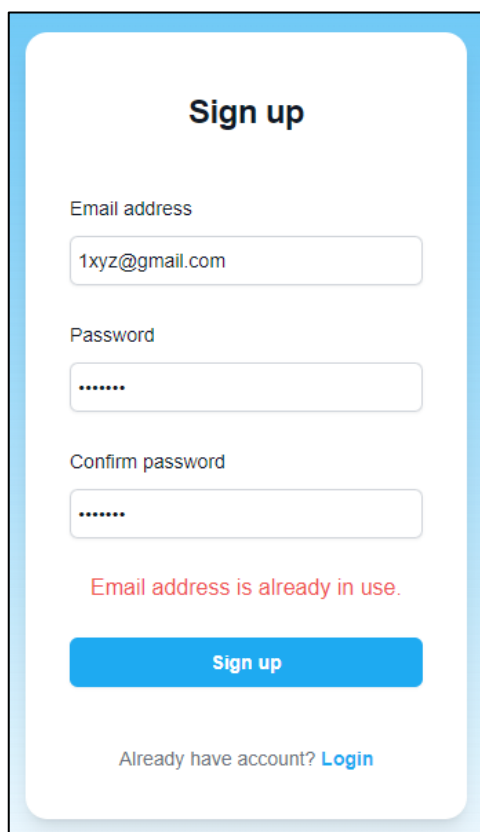
Rys. 5.13. Widok formularza logowania



The image shows a web page for HOACC with a light blue gradient background. In the top left corner, the text "HOACC" is displayed in a bold, white, sans-serif font. Centered on the page is a white rectangular sign-up form with rounded corners and a subtle drop shadow. The form is titled "Sign up" in a bold, black font. It contains three input fields: "Email address", "Password", and "Confirm password", each with a light gray border and a small downward arrow on the right. Below these fields is a solid blue button with the text "Sign up" in white. At the bottom of the form, there is a link that says "Already have account? Login".

Rys. 5.14. Widok formularza rejestracji

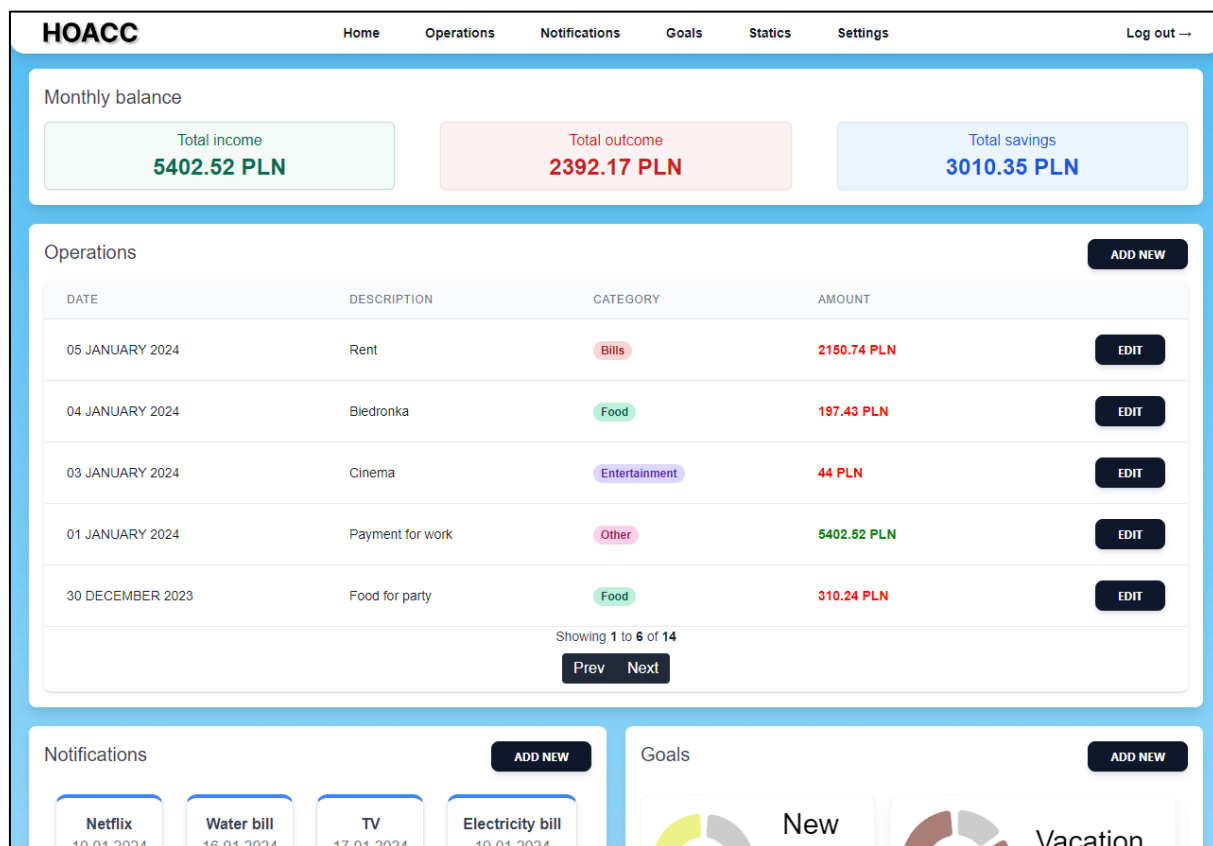
Na rys. 5.15 pokazany jest przykładowy komunikat informujący użytkownika o napotkanym błędzie systemu.



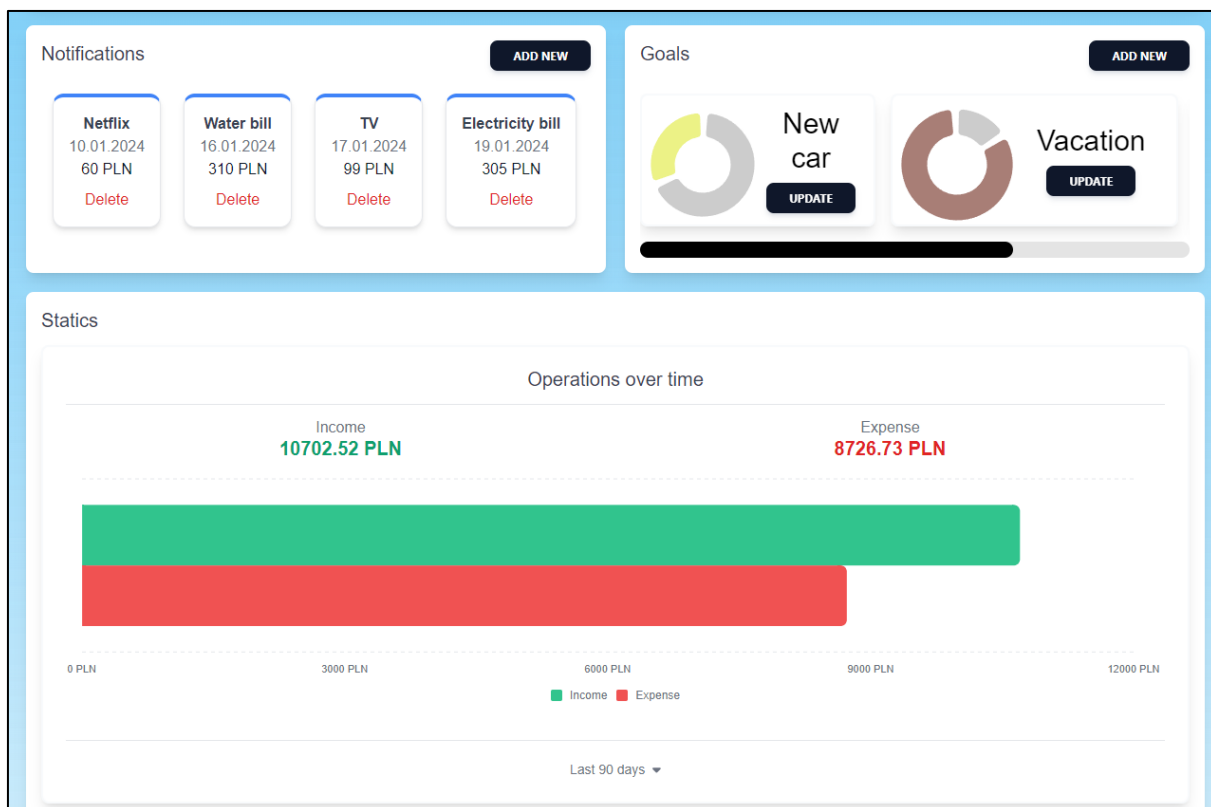
This image shows the same sign-up form as in the previous figure, but with an error message. The "Email address" field now contains the text "1xyz@gmail.com". The "Password" and "Confirm password" fields are filled with seven dots. A red error message, "Email address is already in use.", is displayed below the "Confirm password" field. The blue "Sign up" button and the "Login" link remain at the bottom of the form.

Rys. 5.15. Przykładowy widok komunikatu błędu

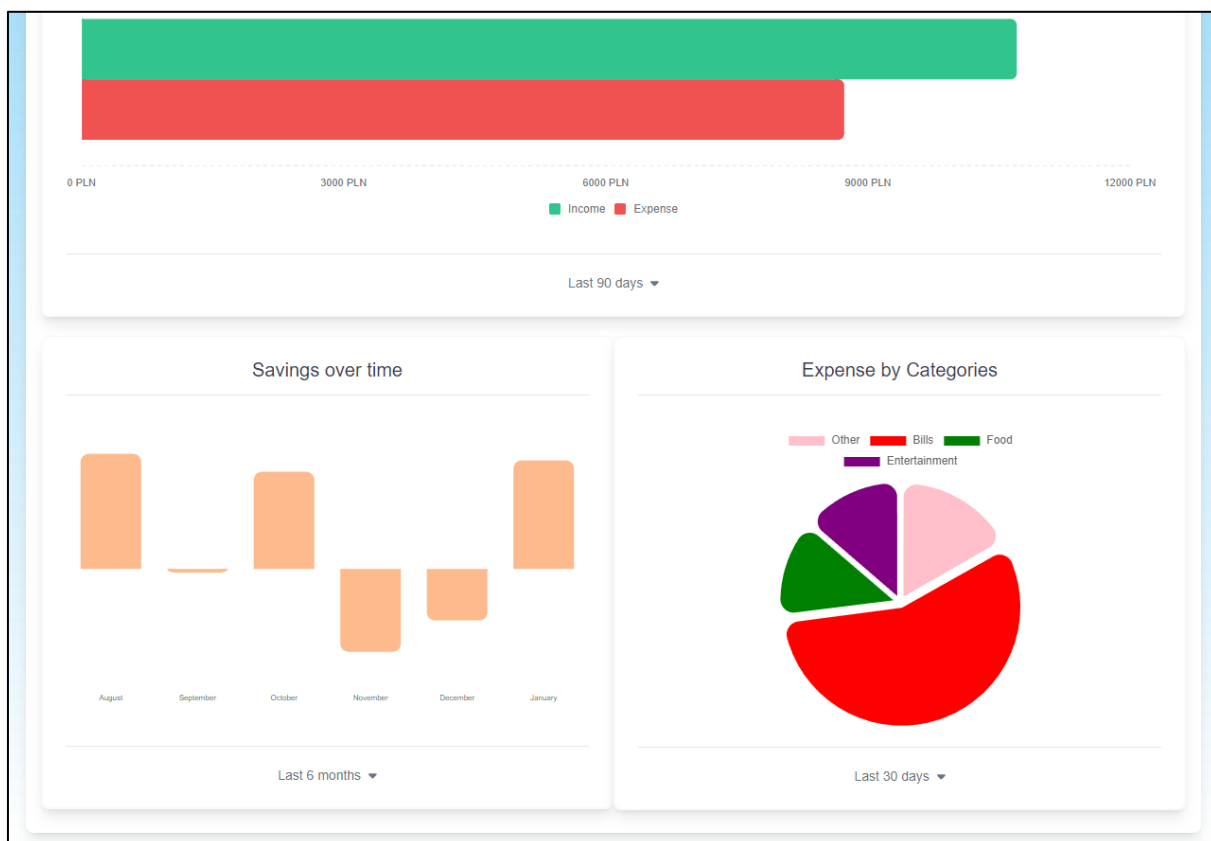
Rys. 5.18-5.20 prezentują panel główny, który dostępny jest po zalogowaniu się do konta użytkownika. Z tego widoku możliwe jest przyjrzenie się sumie przychodów, wydatków oraz oszczędności z bieżącego miesiąca. Użytkownik ma również możliwość obserwacji wszystkich dotychczasowych operacji posortowanych według daty, powiadomień, celów. Ten widok umożliwia także analizę swoich wydatków i przychodów na wykresach. Wykres słupkowy poziomy odpowiedzialny za analizę operacji w czasie oraz wykres kołowy odpowiedzialny za wyświetlanie danych odnośnie wydatków według kategorii ma możliwość wyboru przedziału czasowego takiego jak: wczoraj, dzisiaj, ostatnie siedem dni, ostatnie trzydzieści dni, ostatnie dziewięćdziesiąt dni, ostatnie sześć miesięcy czy ostatni rok co zostało przedstawione na rys. 5.21.



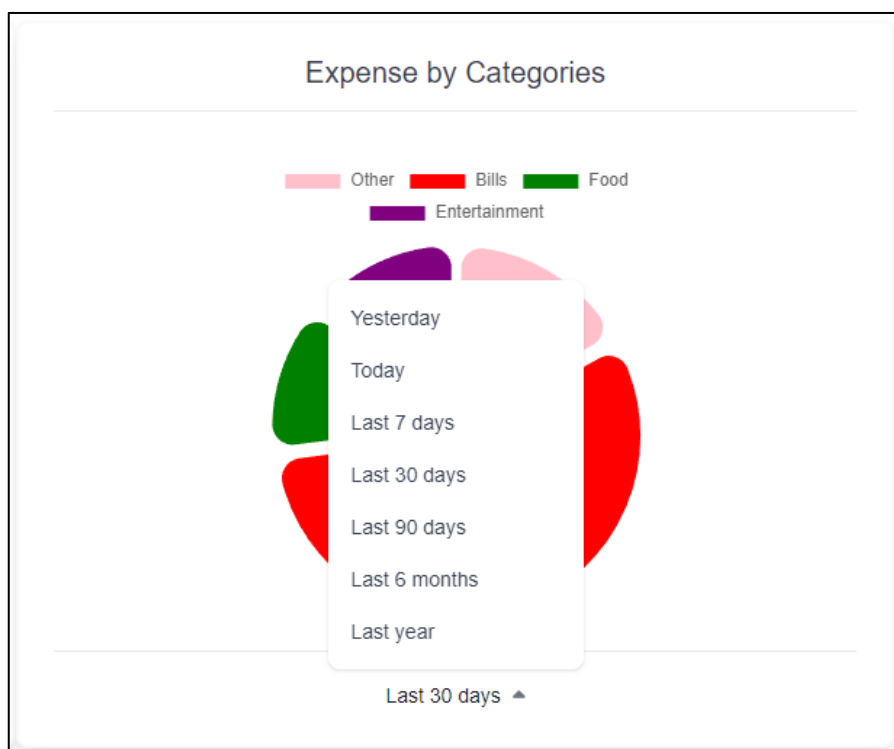
Rys. 5.18. Widok panelu głównego, który dostępny jest po zalogowaniu się do konta użytkownika



Rys. 5.19. Widok panelu głównego

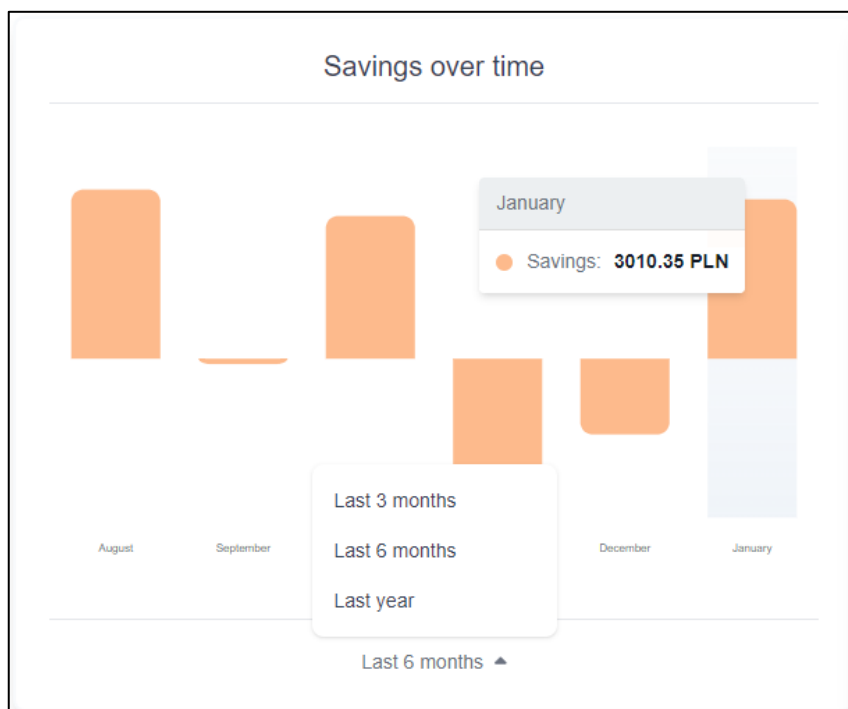


Rys. 5.20. Widok panelu głównego



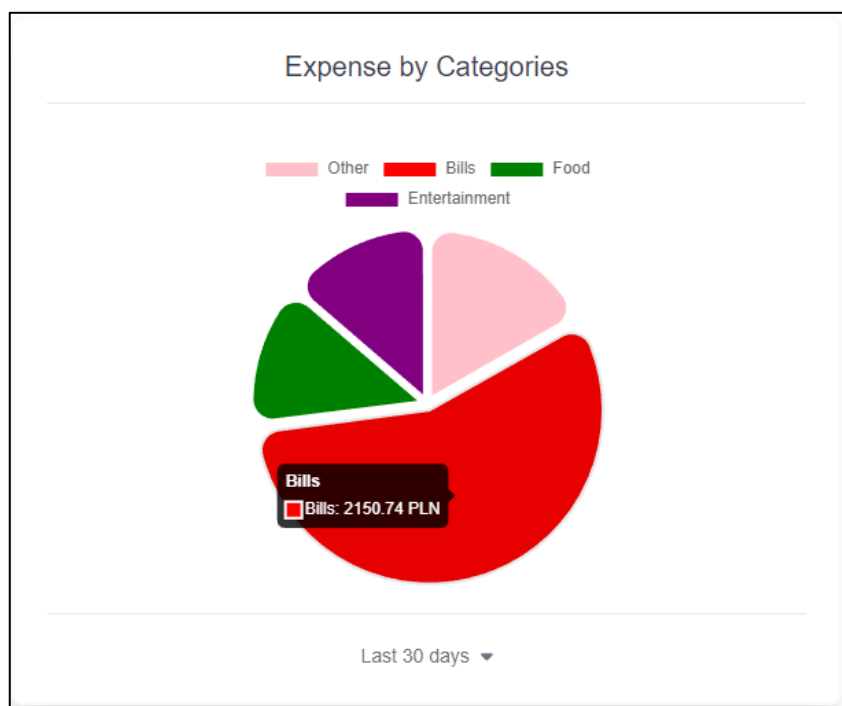
Rys. 5.21. Widok wykresu kołowego ilustrującego wydatki według kategorii z rozwiniętym menu wyboru przedziału czasowego

Natomiast rys. 5.22 ukazuje widok wykresu słupkowego ilustrującego oszczędności w czasie z rozwiniętym menu wyboru przedziału czasowego, użytkownik do wyboru ma trzy opcje: ostatnie trzy miesiące, ostatnie sześć miesięcy i ostatni rok. Ten rysunek ilustruje również wyświetlenie okna modalnego ze szczegółowymi danymi dotyczącymi sumy oszczędności po najechaniu kursorem na wybrany miesiąc.



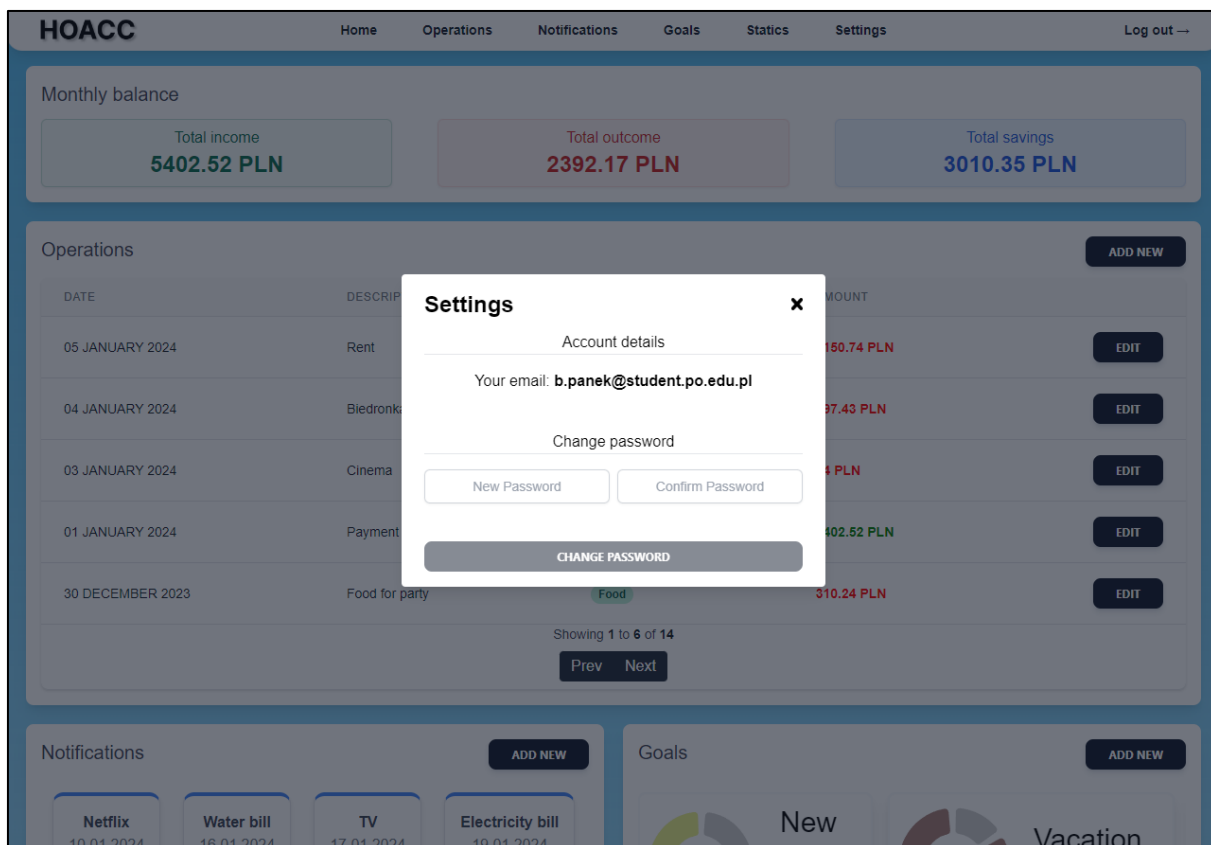
Rys. 5.22. Widok wykresu słupkowego ilustrującego oszczędności w czasie z rozwiniętym menu wyboru przedziału czasowego oraz najechnym kursorem na wybrany miesiąc

Wyświetlanie szczegółowych informacji również dostępne jest dla wykresu operacji w czasie oraz wykresu wydatków według kategorii, ukazuje to rys. 5.23.



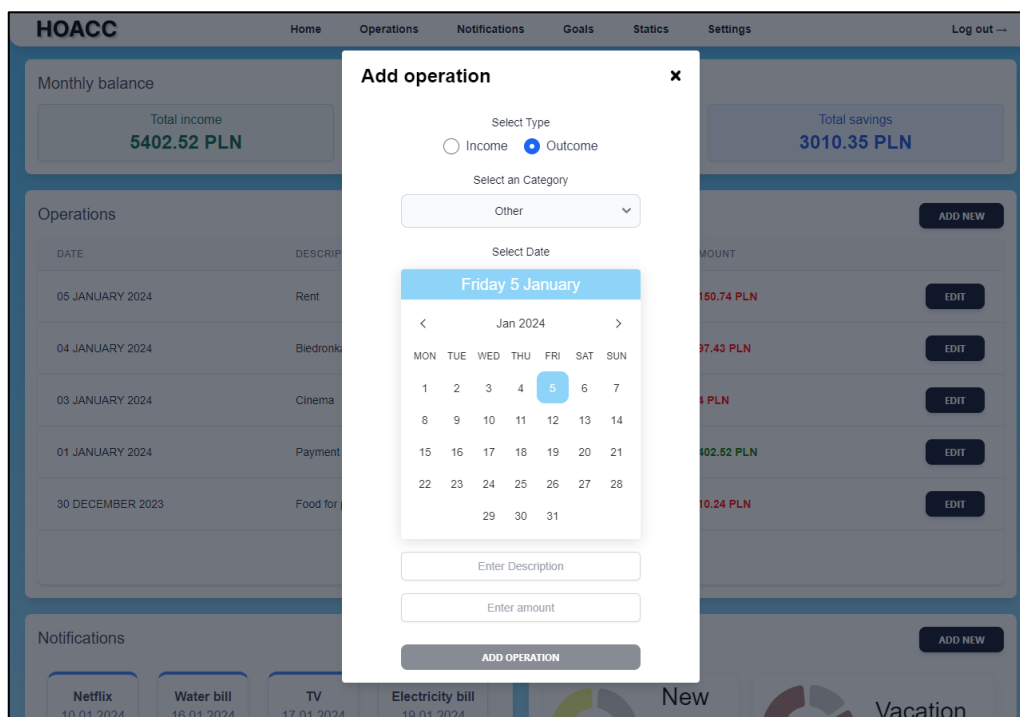
Rys. 5.23. Widok wykresu kołowego ilustrującego wydatki według kategorii podczas najechania kursorem na wybraną kategorię

Panel nawigacji znajdujący się na górze strony w panelu głównym aplikacji po zwała na szybką nawigację pomiędzy elementami aplikacji oraz na wylogowanie się użytkownika z konta za pomocą przycisku „Log out”, po jego naciśnięciu użytkownik zostaje wylogowany i przeniesiony do strony startowej. Na panelu nawigacyjnym istnieje również przycisk „Settings”, który po naciśnięciu otwiera okno modalne z ustawieniami użytkownika, w których może on podejrzeć aktualny adres email zalogowanego konta oraz może zmienić swoje hasło, implementacje ilustruje rys. 5.25.



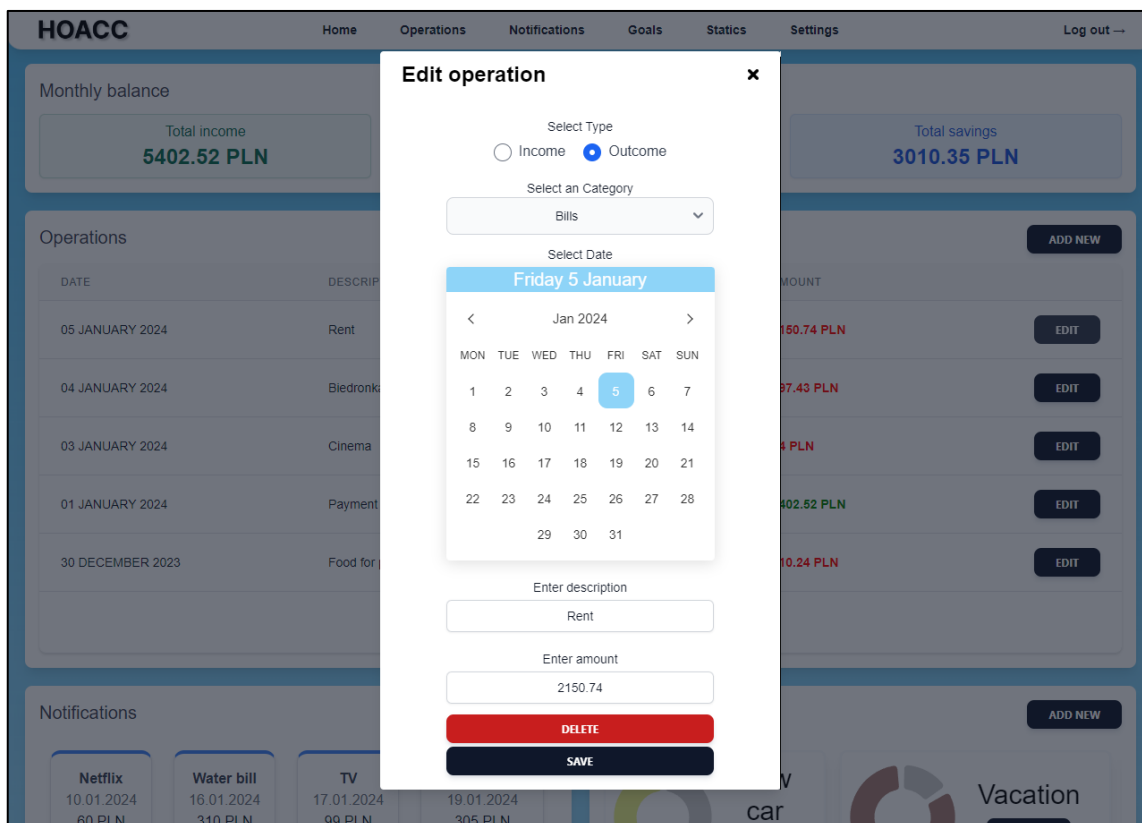
Rys. 5.25. Widok okna modalnego ustawień

Dostęp do widoku panelu głównego umożliwia także dodanie i edycję operacji, okna modalne umożliwiające powyższe operacje dostępne są po naciśnięciu przycisku „ADD NEW” lub „EDIT” dla poszczególnych operacji w sekcji „Operations” na panelu głównym. Okna modalne są widoczne na rys. 5.26 oraz rys. 5.27.



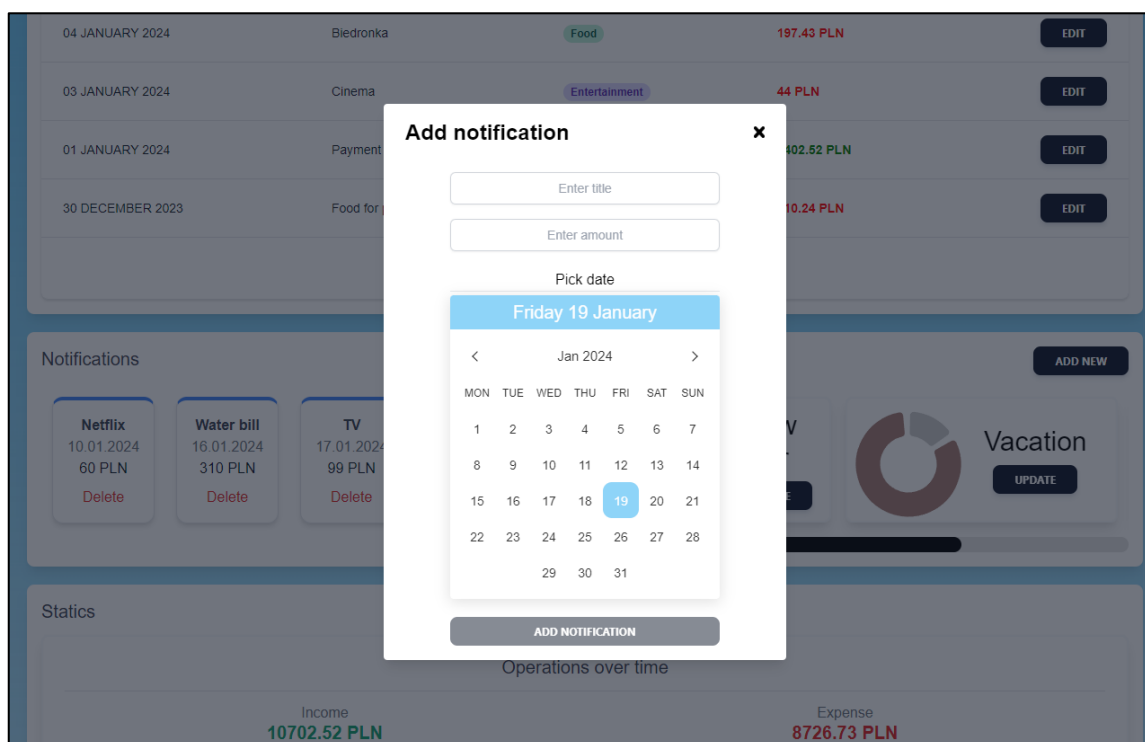
Rys. 5.26. Widok okna modalnego dodawania operacji



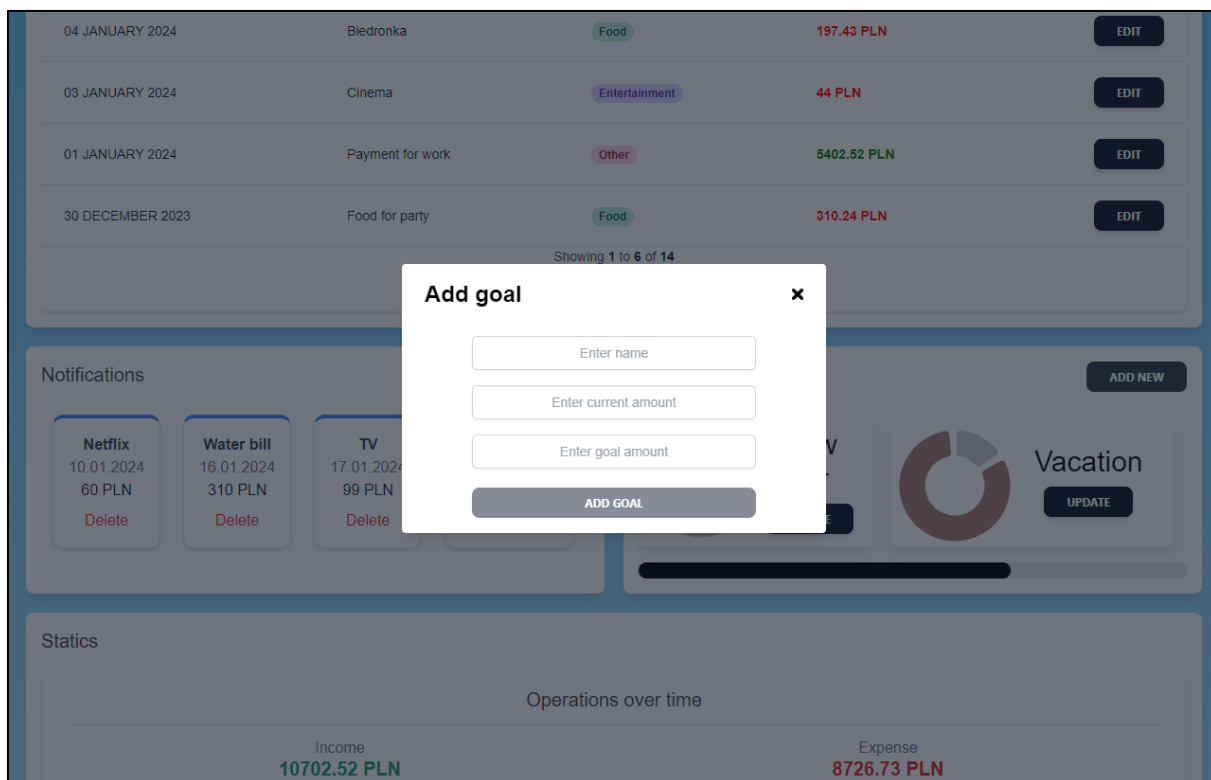


Rys. 5.27. Widok okna modalnego edytowania operacji

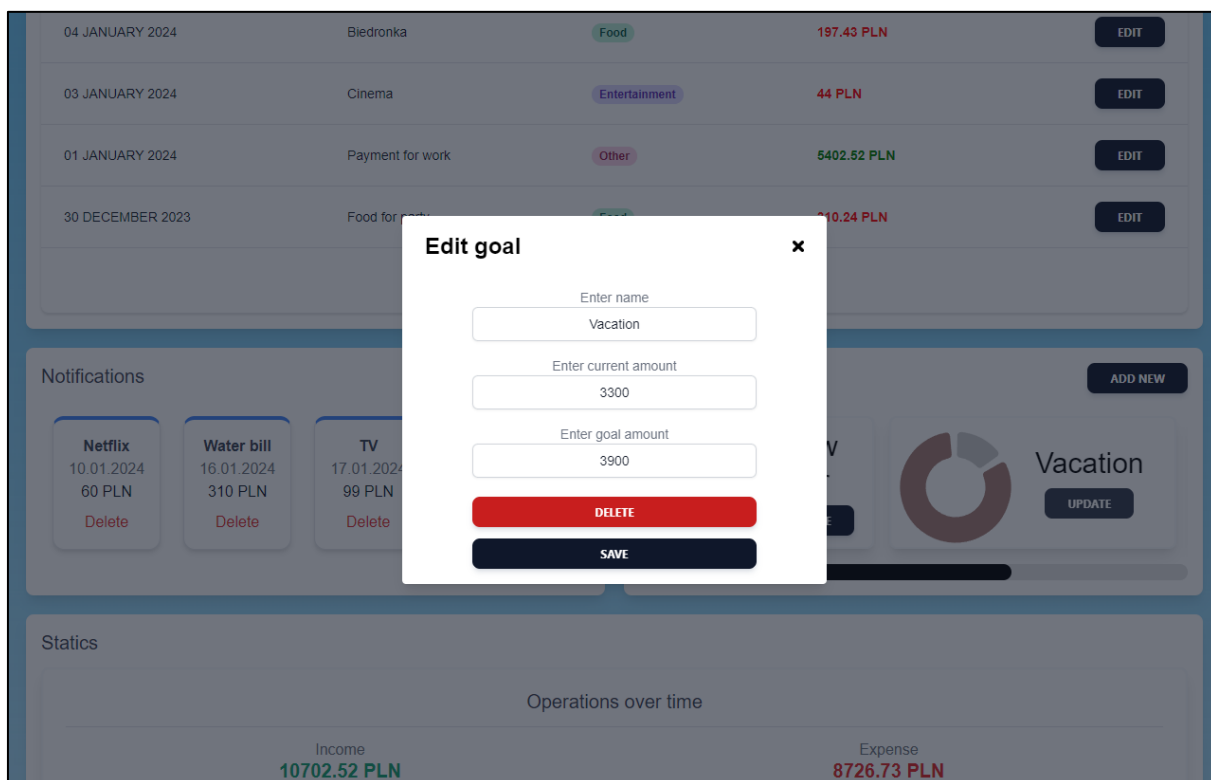
Rysunki 5.29-5.31 kolejno przedstawiają implementacje okna modalnego umożliwiającego dodanie przypomnienia po naciśnięciu przycisku „ADD NEW” w sekcji „Notifications”, dodanie celu po naciśnięciu przycisku „ADD NEW” w sekcji „Goals”, oraz edycję celu po naciśnięciu przycisku „UPDATE” w sekcji „Goals” dla poszczególnego celu.



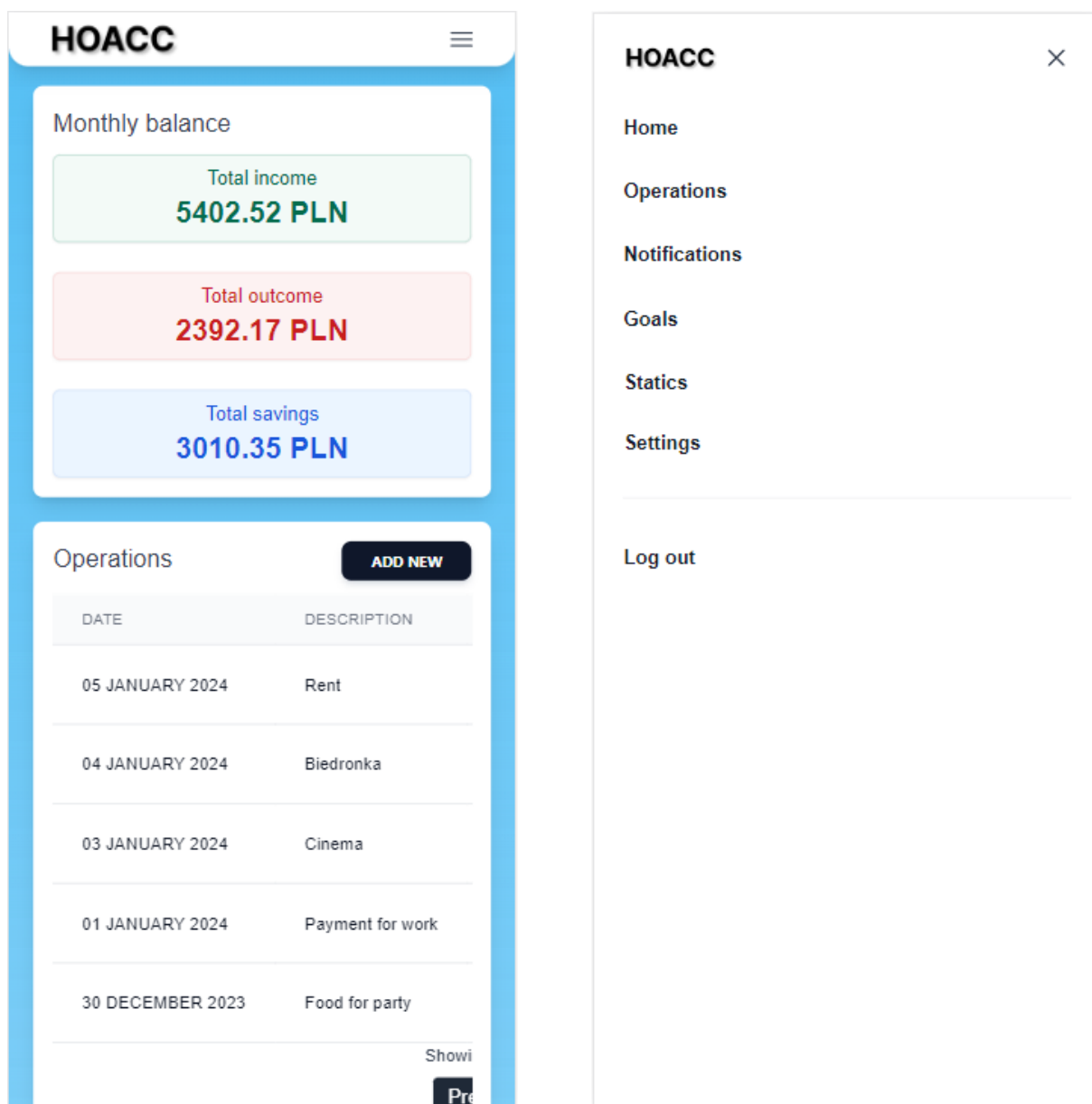
Rys. 5.29. Widok okna modalnego dodawania przypomnienia



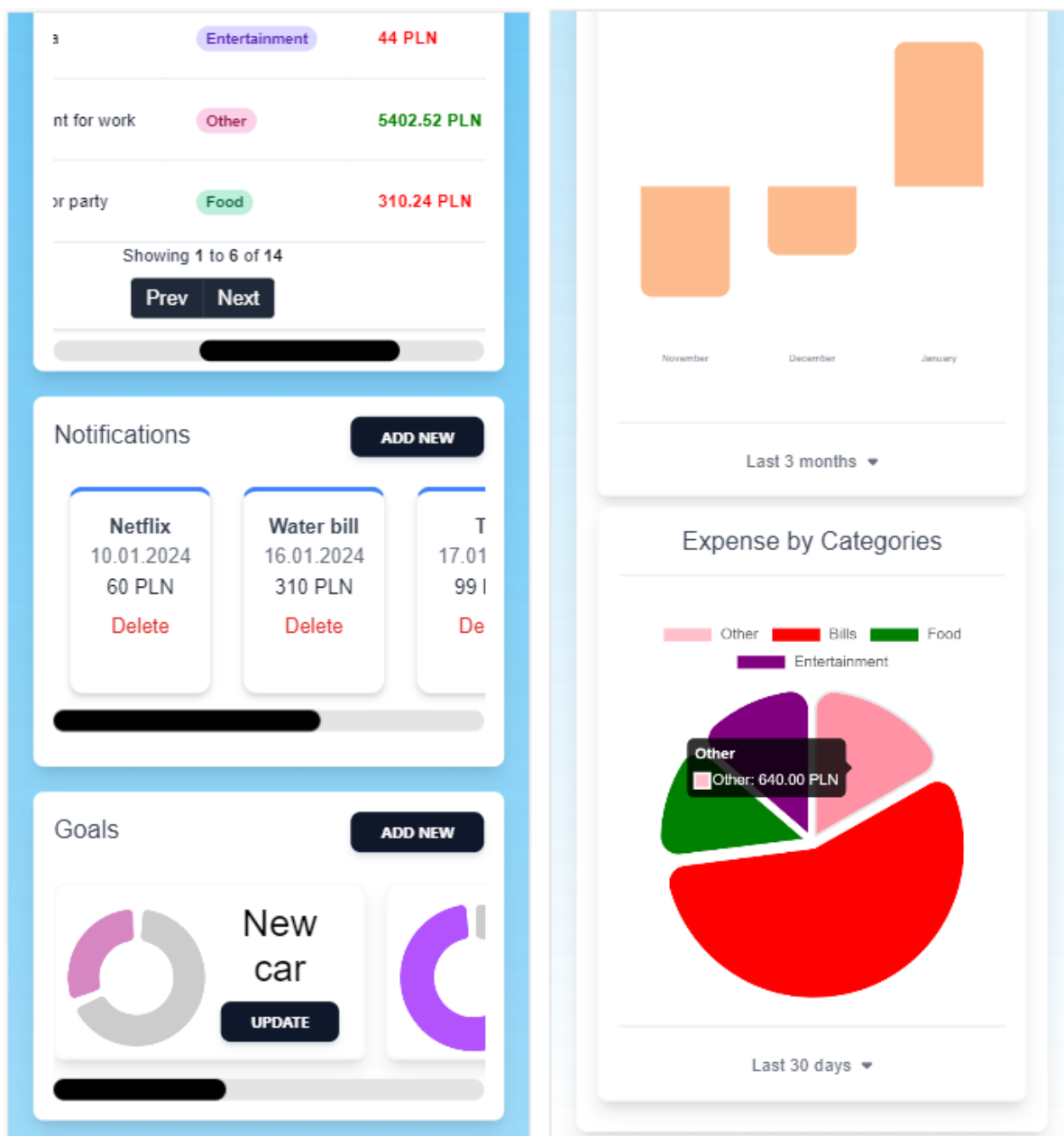
Rys. 5.30. Widok okna modalnego dodawania celu



Rys. 5.31. Widok okna modalnego edytowania celu



Rys. 5.32. Widok interfejsu aplikacji na urządzeniu iPhone XR



Rys. 5.33. Widok interfejsu aplikacji na urządzeniu iPhone XR

Stworzony interfejs został zaprojektowany oraz zaimplementowany z uwzględnieniem responsywności. Oznacza to, że zawartość aplikacji płynnie dostosowuje się do różnych rozmiarów ekranu, co zostało przedstawione na rys. 5.32 oraz rys. 5.33, które ukazują responsywność aplikacji prezentując jej interfejs na urządzeniu mobilnym iPhone XR.