

UNIwersytet WarMińsko–MAZURSKI W OLSZTYNIE

WYDZIAŁ MATEMATYKI I INFORMATYKI

ALEKSANDER CHOJNOWSKI

Informatyka

Aplikacja webowa do zarządzania budżetem domowym

Praca inżynierska wykonana w Katedrze

Metod Matematycznych Informatyki

pod kierunkiem dr. Pawła Drozda

Olsztyn 2023

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

ALEKSANDER CHOJNOWSKI

Computer Science

Web application for managing household budget

**Engineering Thesis written in Chair of
Mathematical Methods of Informatics**

Under supervision of dr Pawel Drozda

Olsztyn 2023

Spis treści

Streszczenie	5
Abstract	6
1. Wstęp	7
2. Opis technologii	9
2.1 Opis bibliotek JavaScript użytych w projekcie React.JS	10
3. Analiza techniczna aplikacji	11
3.1 Słownik pojęć systemowych	11
3.2 Schemat bazy danych	12
3.2.1 Opis poszczególnych tabel bazy danych	12
3.3 Analiza przypadków użycia w systemie	13
3.3.1 Diagram przypadków użycia	14
3.3.2 Opis przypadków użycia	14
3.3.3 Rozwinięcie opisu przypadku użycia „Autoryzuj użytkownika”	16
3.3.4 Rozwinięcie opisu przypadku użycia „Zarządzaj kontem”	19
3.3.5 Rozwinięcie opisu przypadku użycia „Zarządzaj budżetami”	21
3.3.6 Rozwinięcie opisu przypadku użycia „Zarządzaj transakcjami”	28
3.3.7 Rozwinięcie opisu przypadku użycia „Zarządzaj notatkami”	32
3.3.8 Rozwinięcie opisu przypadku użycia „Utwórz raport miesięczny”	36
3.3.9 Rozwinięcie opisu przypadku użycia „Zarządzaj kategoriami”	37
4. Przewodnik po poszczególnych sekcjach i ekranach aplikacji	40
4.1 Autoryzacja użytkownika	40
4.2 Ekran główny aplikacji	41
4.3 Formularz zmiany nazwy użytkownika	43
4.4 Formularz tworzenia nowego budżetu użytkownika	43
4.5 Formularz dodawania użytkownika do budżetu	44
4.6 Lista użytkowników budżetu	44
4.7 Tworzenie nowej transakcji	45

4.8 Widok pojedynczej notatki	46
4.9 Edycja istniejącej transakcji	47
5. Testowanie aplikacji	48
5.1 Zadania testujące	48
5.1.1 Zadanie nr 1	48
5.1.2 Zadanie nr 2	48
5.1.3 Zadanie nr 3	49
5.1.4 Zadanie nr 4	49
5.1.5 Zadanie nr 5	49
5.2 Wnioski płynące z testowania aplikacji	49
6. Podsumowanie i wnioski	50
Bibliografia	51
Spis obrazów	51
Spis tabel	52
Spis diagramów	52

Streszczenie

Celem niniejszej pracy inżynierskiej jest zaimplementowanie responsywnej aplikacji webowej, umożliwiającej zarządzanie budżetem domowym, a także monitorowanie wydatków i przychodów użytkownika, wraz z analizą techniczną zaprojektowanych funkcjonalności. Zadaniem systemu jest umożliwienie użytkownikowi tworzenia wielu Budżetów, w których będzie on mógł zapisywać swoje transakcje, co ma pomóc kontrolować obrót pieniądza w życiu codziennym. Ponadto do każdego z Rachunków daje się przypisać wielu członków, dzięki czemu można wspólnie zarządzać gotówką/zgromadzonymi środkami. Głównym założeniem autora było stworzenie nie tylko przejrzystego, lecz także estetycznego, a przede wszystkim intuicyjnego interfejsu, zachęcającego do korzystania z systemu i przyciągającego nowych odbiorców. W tym celu przeanalizowane zostały potrzeby klientów, a następnie powstały szczegółowe makiety, które przed wdrożeniem zostały przetestowane przez niezależnych testerów. Każda funkcja aplikacji została odpowiednio opisana i zaprezentowana za pomocą diagramów przypadków użycia oraz scenariuszy, które w sposób szczegółowy określają sposób działania danego fragmentu systemu. Praca zawiera opis technologii użytych przy tworzeniu projektu oraz przewodnik po stworzonej aplikacji. Nie zabrakło także miejsca na testy, w których niezależna grupa osób została poproszona o korzystanie z systemu przez siedem dni, a ich opinie na temat działania programu zostały spisane w końcowych rozdziałach.

Abstract

The aim of the engineering thesis is implementation of web application, that allows household budget management, user's income and outcome monitoring, with the technical analysis of designed functionalities. The main task was to enable the user to create many Budgets, where it will be possible to write down all transactions, so the flow of money in everyday life will be controlled. There will be a possibility of adding multiple members to one group, in effect to help managing common funds. The goal was to create clear, aesthetic and above all intuitive user interface, that encourages use of the system and attracts new recipients. For this purpose customer needs have been analysed and detailed mockups have been created, that were being tested by independent testers before deployment. Every function of the application was properly described and presented with use of the use case diagrams and scenarios, showing detailed mode of action of specific system part. Thesis also includes a description of web technologies used during the creation of the project, as well as an application guide. There was no shortage for tests, where independent group of people was asked to use the system for seven days, and their opinions about program's functions were written down in the ending chapters.

1. Wstęp

Tematem niniejszej pracy inżynierskiej jest aplikacja webowa do zarządzania budżetem domowym. Codzienne wydatki, a także nieustanny (oraz nierzadko nieumiejętny) obrót pieniędzmi może spowodować powstanie problemów na tle ekonomicznym, choć zazwyczaj rozpoczyna się od zdumienia bądź konsternacji, gdy okazuje się, że na koniec miesiąca w portfelu zostało znacznie mniej, niż przypuszczaliśmy. Niekontrolowanie swoich wydatków i przychodów oraz sposobu zarządzania środkami sprawia, że ludzie są zdecydowanie bardziej skłonni do poddania się konsumpcjonistycznemu pędowi do nabywania produktów zbędnych.

Prezentowana/opisana w pracy Aplikacja została stworzona w odpowiedzi na realne potrzeby ludzi wokół, jak i samego twórcy. To właśnie dzięki niej użytkownik – po zrobieniu zakupów – może wziąć telefon komórkowy do ręki, wejść na przeznaczoną do tego stronę w wyszukiwarce i zarejestrować wydatek w odpowiednim budżecie, czyli swoistej kategorii, służącej do zapisywania i monitorowania transakcji gotówkowych. Aplikacja umożliwia również rejestrowanie w systemie transakcji przychodzących, przydatnych w momencie, gdy użytkownik wejdzie w posiadanie nowych pieniędzy. Co więcej, budżety użytkowników mogą być zarządzane przez wielu członków symultanicznie/jednocześnie, dzięki czemu w obrębie jednego gospodarstwa domowego wszyscy mogą kontrolować i monitorować wydatki. W aplikacji nie zabrakło także miejsca na notatki użytkownika, gdzie można zapisywać najważniejsze informacje lub, nieco bardziej prozaicznie, sporządzić listę zakupów.

Ważnym aspektem systemu są definiowane przez administratora kategorie transakcji, przy użyciu których użytkownik aplikacji może w dokładny sposób zdefiniować, jaki charakter miał wydatek czy przychód, między innymi są to opcje takie jak: „Artykuły domowe”, „Artykuły Spożywcze”, „Wypłata”, „Znalezione” i wiele innych.

Dzięki rzetelnemu zapisywaniu/spisywaniu/notowaniu wszystkich transakcji, jakich dokonywali członkowie podłączeni do danego budżetu, oraz możliwości podziału wszelkich zdarzeń przy pomocy konkretnych kategorii, użytkownik może wygenerować miesięczny raport zawierający najważniejsze dane z konkretnego miesiąca, przedstawione w formie czytelnych wykresów: funkcja ta pomaga w codziennym monitorowaniu stanu finansowego budżetu i pomaga uświadomić użytkownikowi, na którą kategorię (a więc na przykład „Rozrywka” bądź „Artykuły spożywcze”) przeznaczają najwięcej środków z budżetu.

System powstał z wykorzystaniem najnowszych i sprawdzonych technologii do tworzenia aplikacji internetowych. Warstwa wizualna została zaprojektowana w darmowym

programie do tworzenia makiet Figma, a w trakcie implementacji zaprogramowana przy użyciu biblioteki JavaScript – React.JS. Główną myślą, jaka przyświecała autorowi przy tworzeniu interfejsów użytkownika, była ich przejrzystość i intuicyjność. System ma zachęcać nowych klientów do korzystania i kontrolowania swoich wydatków.

Warstwa niewidoczna dla użytkownika (naturalnie nie mniej ważna w funkcjonowaniu aplikacji), czyli łączenie się z bazą danych i wysyłanie zapytań w stronę serwera, została stworzona w języku programowania Python, głównie z wykorzystaniem biblioteki FastAPI, a także bibliotek pomocniczych, takich jak choćby: SQLAlchemy do łączenia się z bazą danych i manipulowania danymi w niej, PyJWT do zabezpieczenia danych użytkownika oraz bcrypt, której zadaniem jest szyfrowanie wrażliwych danych użytkownika.

2. Opis technologii

HTML (eng. HyperText Markup Language) – język znaczników wykorzystywany do modelowania struktury aplikacji webowych oraz stron internetowych. Przy pomocy HTML programista jest w stanie uzyskać schemat, w jakim dane zostaną wyświetlone. W skład języka wchodzi znaczniki określające cel ich zastosowania, na przykład `<input/>` używany jest, gdy oczekuje się od użytkownika wprowadzenia określonych danych.

CSS (eng. Cascading Style Sheets) – lista reguł, którymi poprzez wprowadzenie określonych wartości, programista manipuluje wyglądem oraz zachowaniem się elementów struktury strony, między innymi poprzez określenie koloru tła, rozmiaru elementu czy marginesu wewnętrznego i zewnętrznego, a także pozycjonowania elementu na stronie. Arkusz stylów zawiera także wiele reguł określanych jako pseudo-elementy, które umożliwiają zmianę wyglądu danego fragmentu strony w konkretnych, specyficznych sytuacjach.

SCSS – jeden z najpopularniejszych pre-procesorów do CSS. Optymalizuje czas i usprawnia pracę, którą należy wykonać podczas definiowania reguł arkusza stylów przez programistę. Kod napisany w SCSS, jest redagowany do formatu `.css`, który przeglądarka jest w stanie odczytać i zastosować do elementów na stronie.

JavaScript – język programowania, który w aplikacjach webowych oraz stronach internetowych wywoływany jest po stronie użytkownika i wykorzystywany do interakcji z elementami struktury dokumentu. Dzięki JavaScript przeglądarka jest w stanie zarejestrować zdarzenia wywołane przez klienta, na przykład kliknięcie na dany element, i wywołanie określonych działań.

React.JS – biblioteka JavaScript, optymalizująca i ułatwiająca tworzenie zaawansowanych projektów oraz aplikacji webowych. React ułatwia sposób zarządzania elementami na stronie oraz danymi otrzymywanymi z serwera, a dobrze zaprojektowana struktura plików poprawia czytelność kodu oraz usprawnia dalszy rozwój aplikacji.

Python – popularny język programowania wysokiego poziomu, najczęściej wykorzystywany do modelowania i wizualizacji danych w uczeniu maszynowym, a także do tworzenia interfejsów aplikacji. Dzięki swojej prostocie i czytelności kodu jest częstym wyborem do nauki programowania przez początkujących programistów. Dodatkowo oferuje bogatą ilość bibliotek rozbudowujących podstawowe funkcje języka.

FastAPI – biblioteka języka Python wykorzystywana do tworzenia interfejsów programistycznych aplikacji, w skrócie API (eng. Application Programming Language). Poprzez swoją prostotę i przejrzystość umożliwia szybkie i sprawne tworzenie zaplecza aplikacji, nawet przez mniej wprawionych programistów.

Swagger UI – narzędzie, które w sposób automatyczny tworzy dokumentację dla utworzonych ścieżek końcowych interfejsu aplikacji, wraz z formularzami umożliwiającymi przetestowanie zaprojektowanych funkcjonalności.

Git – najpopularniejszy system kontroli wersji, umożliwiający dokumentowanie postępów pracy nad projektem. Jest także w pewnym sensie zabezpieczeniem kodu, gdyż poprzez odpowiednie komendy programista jest w stanie przywrócić wcześniejszy kod, na przykład wersję z działającym kodem, gdy aktualny nie działa.

GitHub – serwis internetowy z możliwością przechowywania repozytoriów z kodem aplikacji bez konieczności opłaty za taką usługę. GitHub wykorzystuje system kontroli wersji GIT, dzięki któremu w łatwy sposób z poziomu wiersza poleceń na komputerze można przesłać pliki na serwer serwisu, gdzie będą przechowywane.

Visual Studio Code – popularny i darmowy edytor kodu źródłowego aplikacji. Ma przejrzysty i intuicyjny interfejs, a także umożliwia formatowanie kodu poprzez kolorowanie składni, co znacznie usprawnia i ułatwia pracę. Połączenie tych elementów z bogatym zbiorem wtyczek i dodatków tworzy świetne narzędzia do pracy z kodem.

2.1 Opis bibliotek JavaScript użytych w projekcie React.JS

Material UI – biblioteka zawierająca pokaźny zbiór ikon i komponentów poprawiających wygląd aplikacji.

Date-fns – biblioteka, której zadaniem jest zmiana standardowego, zwracanego przez serwer oraz elementy na stronie, formatu daty i godziny na format czytelny dla użytkownika. Na przykład format „2022-01-23T12:34” Date-fns przeformatuje na „wczoraj o 12:34”.

Chart.js – biblioteka języka JavaScript umożliwiająca wizualizację danych na stronie internetowej. Otrzymawszy odpowiednio sformatowane dane, wyświetla je w formie wykresu, a dzięki zdefiniowaniu konkretnych opcji jest w stanie animować je i poprawiać ich czytelność.

Sweetalert2 i Toastify – biblioteki, których głównym zadaniem jest tworzenie i wyświetlanie w przejrzysty sposób informacji w formie alertów i pop-upów.

3. Analiza techniczna aplikacji

Aplikacje webowe, systemy informatyczne i wszelkie inne oprogramowanie, będące w przyszłości obsługiwane przez wielu użytkowników w różnym przedziale wiekowym, z odmiennymi zdolnościami obsługi komputera oraz niejednorodnym wykształceniem, powinny być odpowiednio przemyślane i przeanalizowane pod kątem biznesowym i intuicyjności interfejsów użytkownika. Dobra analiza i przejrzysta dokumentacja poszczególnych funkcjonalności systemu, to nie tylko wygoda użytkownika w korzystaniu z aplikacji, lecz także łatwiejszy sposób na wdrożenie nowych programistów w skład zespołu, który stanie za rozwojem programu. Obecnie istnieje wiele narzędzi, dzięki którym analityk jest w stanie przedstawić, w jaki sposób działają poszczególne sekcje i funkcjonalności. W poniższej analizie zastosowane zostaną instrumenty takie jak diagramy przypadków użycia w formie ogólnej i szczegółowej, a także scenariusze poszczególnych funkcji użytkownika i administratora.

3.1 Słownik pojęć systemowych

Użytkownik / klient – jednoznacznie rozróżnialne konto w systemie identyfikujące się swoim unikalnym numerem ID. Ponadto każdy użytkownik posiada unikalny login, którym z wykorzystaniem hasła składającego się minimum z 8 znaków, autoryzuje się w aplikacji.

Rola użytkownika – każde konto w systemie posiada jedną z dwóch ról: użytkownik lub administrator. Rola określana jest przez tabelę 'isStaff' w bazie danych.

Administrator – użytkownik posiadający uprawnienia do tworzenia, modyfikacji oraz usuwania kategorii. Administrator posiada dostęp do wszelkich niewrażliwych danych zawartych w systemie.

Transakcja – pojedynczy obiekt przypisany jednoznacznie do konkretnego budżetu oraz użytkownika. Zawiera informacje o konkretnym obrocie pieniądza w realnym świecie.

Notatka – obiekt systemu jednoznacznie przypisany do określonego budżetu. Notatka zawiera treść zapisaną przez jednego z użytkowników danego rachunku.

Kategoria – encja w aplikacji zawierająca nazwę, unikalne ID oraz informację określającą typ transakcji.

Budżet/Rachunek – jednoznacznie identyfikowalny obiekt systemu, składający się z listy transakcji, notatek oraz osób, które mogą nimi zarządzać.

3.2 Schemat bazy danych

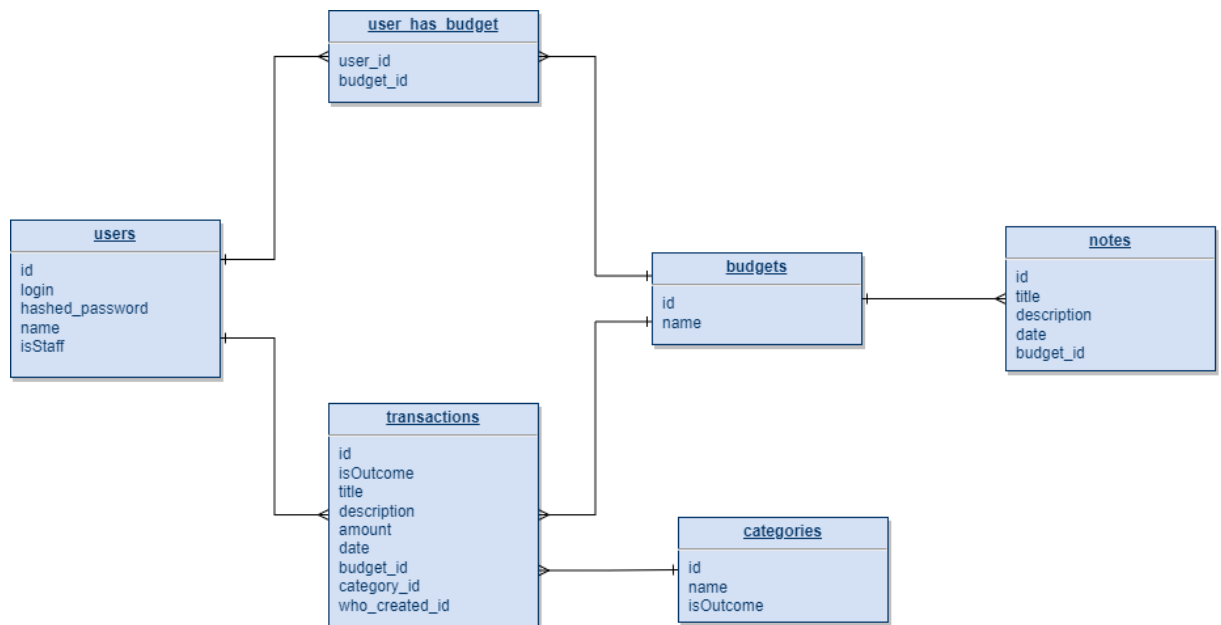


Diagram 1 Schemat bazy danych

3.2.1 Opis poszczególnych tabel bazy danych

Users – każdy użytkownik systemu posiada swoje unikalne ID, które może publicznie udostępniać innym klientom aplikacji. Login oraz hasło stanowią niezbędne dane potrzebne do autoryzacji użytkownika, a pole ‘name’ zawiera tekst, który ma się wyświetlać przy transakcjach danej osoby oraz w panelach aplikacji. Zaleca się, aby dane logowania, tj. login oraz hasło, pozostały prywatne dla właściciela konta w celach bezpieczeństwa, natomiast informacje zawierające się w polu ‘name’ nie są danymi wrażliwymi i mogą być publicznie udostępniane.

Budgets – budżety użytkowników posiadają unikalne ID oraz nazwę, która może się powtarzać. Oznacza to, że mimo istnienia dwóch budżetów o identycznej nazwie, są one innymi obiektami ze względu na ich różne ID.

User_has_budget – tabela zawierająca informację o tym, że użytkownik o konkretnym ID jest na liście dostępu do budżetu o danym ID, na przykład można przyjąć, że oznaczenie 5:16 to nic innego, jak określenie, że osoba o ID 5 jest na liście użytkowników budżetu o numerze ID 16.

Transactions – każda transakcja w tabeli składa się z tytułu, kategorii, kwoty, oraz daty zawarcia, opcjonalnie może mieć także krótki opis. Każda transakcja ma jeden z dwóch typów: „Wydatek” lub „Przychód”, które określane są przy pomocy kolumny isOutcome przyjmującej wartość odpowiednio 1 lub 0.

Categories – kategoria posiada swoje unikalne ID, tytuł oraz przeznaczenie, tj. czy może być użyta do transakcji o charakterze „Wydatku” czy „Przychodu”.

Notes – notatka jest jednoznacznie przypisana do konkretnego budżetu za pośrednictwem klucza obcego budget_id. Każdy obiekt składa się z tytułu, opisu oraz daty, a w systemie identyfikowany jest przez unikalny numer id.

Raport miesięczny – zbiór informacji zawierający dane transakcji z konkretnego miesiąca i budżetu, przedstawiony w formie wykresów. Raport przedstawia dane takie jak: sumę wszystkich „Wydatków” i „Przychodów” oraz saldo budżetu, sumę wydatków i przychodów jaką każdego użytkownika wykazaną w danym miesiącu, podział transakcji na konkretne kategorie i ich procentowy udział.

3.3 Analiza przypadków użycia w systemie

Jednym z najbardziej przejrzystych i intuicyjnych sposobów na przedstawienie możliwości danej grupy użytkowników systemu jest zaprezentowanie tego w formie graficznego diagramu przypadków użycia. Diagram ten składa się z aktorów, którymi są dane grupy użytkowników lub inne nieożywione podmioty, na przykład system płatności, oraz z przypadków użycia (eng. Use Case), które połączone są między sobą i aktorami za pomocą związków asocjacyjnych i przedstawiają daną funkcjonalność w systemie. Przyszły system został skrupulatnie przeanalizowany, a wynikające z tego udogodnienia użytkownika zostały zaprojektowane w darmowym programie draw.io.

3.3.1 Diagram przypadków użycia

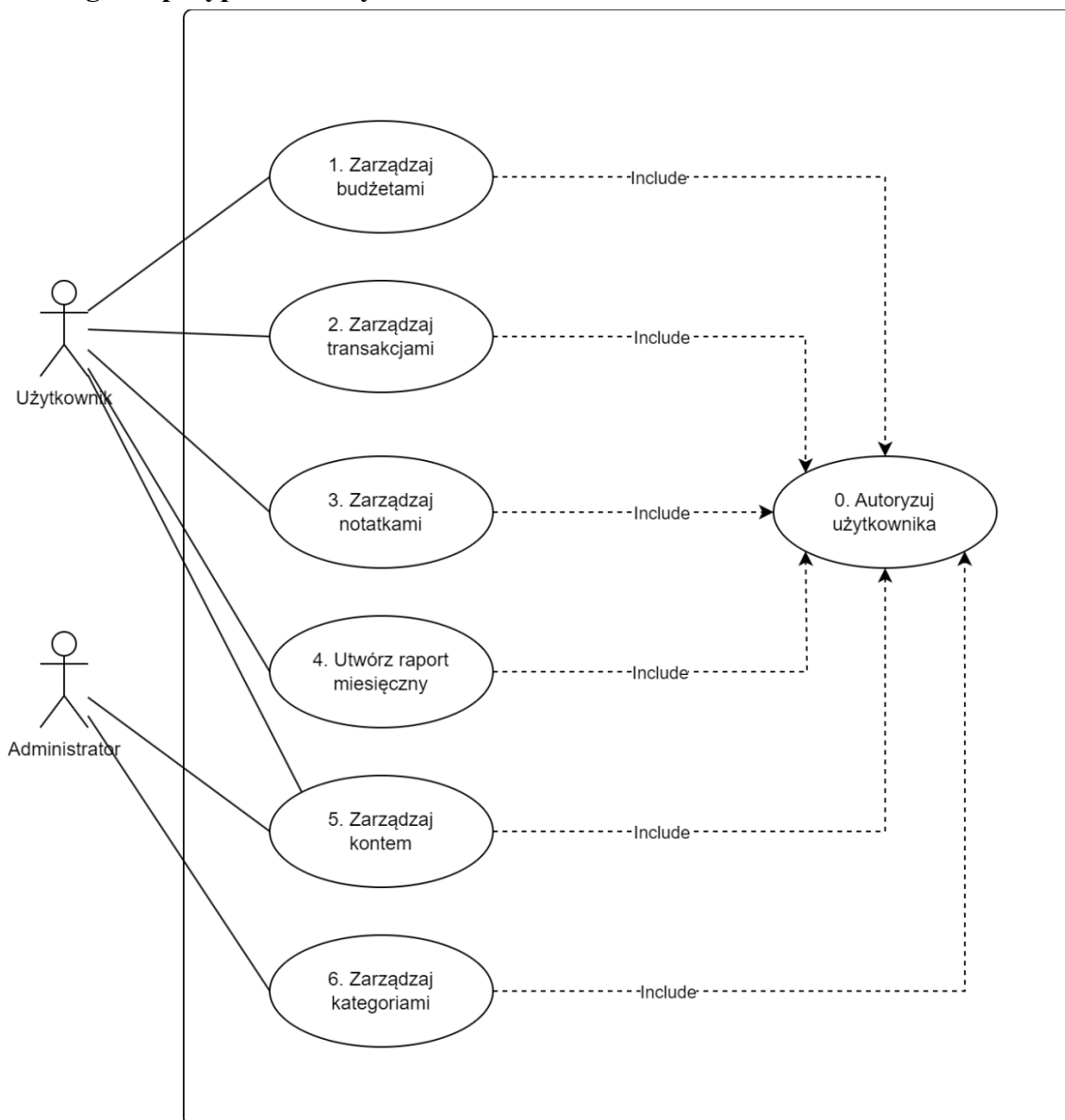


Diagram 2 Główny diagram przypadków użycia

3.3.2 Opis przypadków użycia

Autoryzuj użytkownika – podstawową, i obowiązkową do dalszych działań w systemie, funkcjonalnością jest autoryzacja użytkownika. Aby system poprawnie zidentyfikował daną osobę musi ona wybrać i przejść jedną z dwóch możliwych ścieżek: zalogować się, jeśli gość posiada już konto w systemie, lub skorzystać z opcji rejestracji, która jest przeznaczona dla osób chętnych do skorzystania z usług aplikacji.

Zarządzaj kontem – funkcja dzięki której użytkownik systemu jest w stanie manipulować swoją wyświetlaną nazwą. Administrator posiada opcję zmiany roli użytkownika z poziomu bazy danych.

Zarządzaj budżetami – zbiór funkcji dla użytkownika dzięki którym jest on w stanie manipulować danymi związanymi z jego rachunkami. Posiadając budżety, klient jest w stanie wybrać konkretny z nich, by następnie zarządzać członkami przypisanymi do danego budżetu poprzez usuwanie lub dodawanie osób. W przypadku braku budżetów przypisanych do konta użytkownika, może on skorzystać z opcji „Dodaj budżet” w celu stworzenia nowego.

Zarządzaj transakcjami – funkcje umożliwiające użytkownikowi dodawanie nowych transakcji do konkretnego budżetu, w którym jest członkiem, a także modyfikacje oraz usuwanie stworzonych wcześniej obiektów tego typu. Po wypełnieniu niezbędnych danych w formularzu, klient może skorzystać z opcji „Stwórz transakcję”, a zostanie ona dodana do budżetu oraz automatycznie wyświetlona na liście transakcji. Transakcje, które zostały utworzone przez zautoryzowanego użytkownika posiadają opcję edycji, a także możliwość usunięcia. Oznacza to, że osoba ma uprawnienia do zarządzania jedynie transakcjami stworzonymi przez siebie.

Zarządzaj notatkami – funkcje, dzięki którym użytkownik aplikacji może dodać nową notatkę do budżetu. Po utworzeniu notatki klientowi udostępnione są kolejne funkcje modyfikacji notatki, takie jak edycja treści oraz możliwość usunięcia jej.

Utwórz raport miesięczny – funkcja, która umożliwia wyświetlenie raportu miesięcznego dla konkretnego budżetu i miesiąca.

Zarządzaj kategoriami – zbiór funkcji dla administratora systemu, umożliwiający dodanie nowej kategorii transakcji, a także modyfikację i usuwanie istniejących już w bazie danych informacji.

3.3.3 Rozwinięcie opisu przypadku użycia „Autoryzuj użytkownika”.

Autoryzacja użytkownika jest najistotniejszą funkcją w całym systemie — to dzięki poprawnemu przejściu procesu logowania lub rejestracji do aplikacji, program wie, kim jest osoba po drugiej stronie monitora lub telefonu komórkowego, a dzięki danym uzyskanym podczas tego procesu jest w stanie wyświetlać odpowiednie treści; w przypadku wysyłania zapytań w formie tworzenia nowych treści na stronie, przypisuje odpowiednie konto do powstałego obiektu.

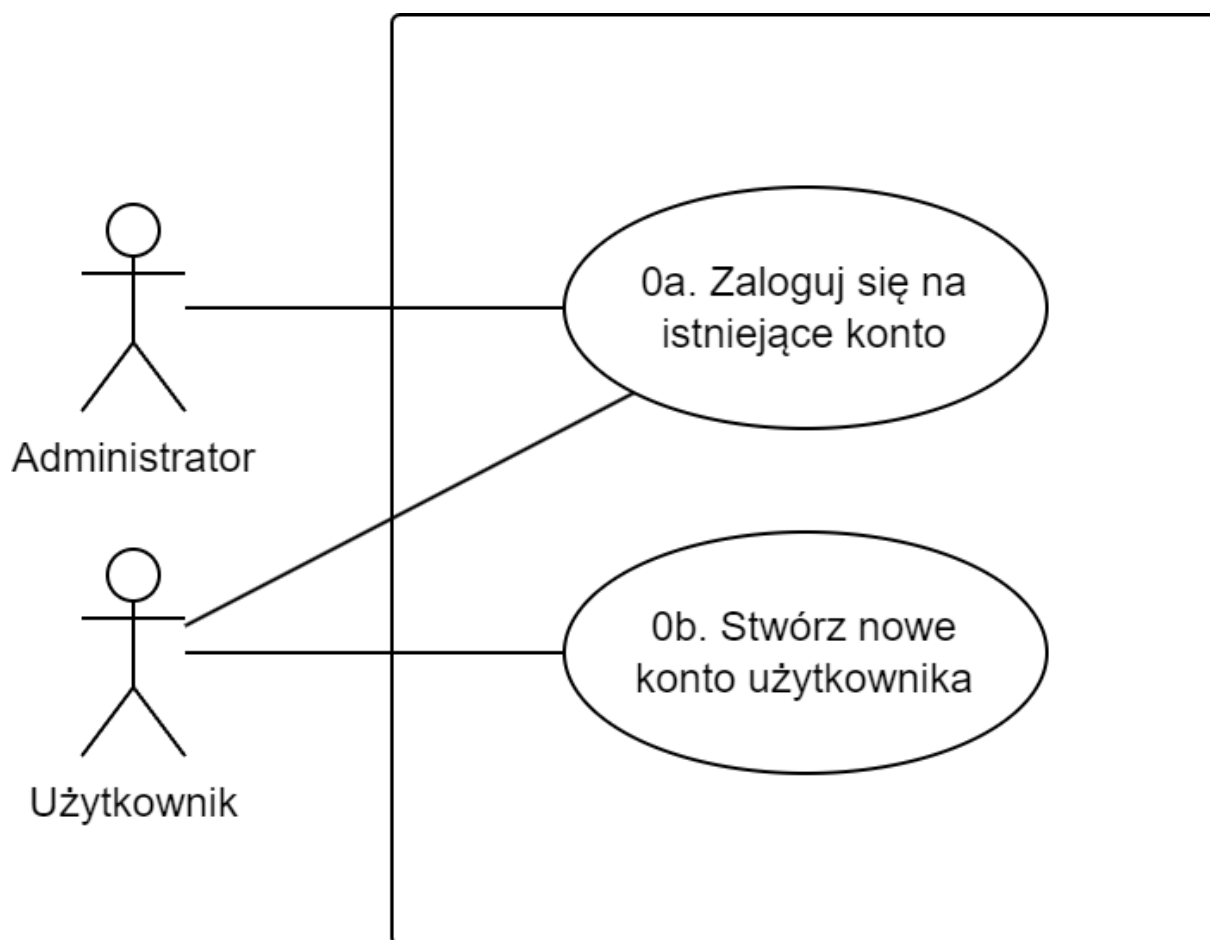


Diagram 3 Szczegółowy diagram przypadku użycia „Autoryzuj użytkownika”

Scenariusz przypadku użycia „Zaloguj się na istniejące konto”

Tabela 1 Scenariusz przypadku użycia „Zaloguj się na istniejące konto”

Identyfikator przypadku użycia	0a
Nazwa przypadku użycia	Zaloguj się na istniejące konto
Cel przypadku użycia	Poprawne przejście procesu logowania do systemu.
Aktorzy	Użytkownik, Administrator
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Użytkownik posiada konto w systemie.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none">1. Użytkownik otwiera aplikację.2. System wyświetla formularz logowania do systemu.3. Użytkownik wypełnia niezbędne dane logowania, tj. login i hasło.4. Użytkownik zatwierdza wprowadzone informacje przyciskiem <Zaloguj>, lub klikając klawisz „ENTER”.5. System sprawdza zgodność loginu z hasłem.6. System autoryzuje użytkownika i przenosi go do ekranu aplikacji. <p><u>Przebiegi alternatywne:</u></p> <ol style="list-style-type: none">6a. System wyświetla komunikat o nieprawidłowych danych logowania.
Efekt końcowy	Poprawne zalogowanie się do systemu i autoryzacja użytkownika.
Uwagi	brak

Scenariusz przypadku użycia „Stwórz nowe konto użytkownika”

Tabela 2 Scenariusz przypadku użycia „Stwórz nowe konto użytkownika”

Identyfikator przypadku użycia	0b
Nazwa przypadku użycia	Stwórz nowe konto użytkownika
Cel przypadku użycia	Poprawne przejście procesu rejestracji nowego konta w systemie.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Brak
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none">1. Użytkownik otwiera aplikację.2. System wyświetla formularz logowania do systemu.3. Użytkownik wyraża chęć założenia konta w systemie klikając przycisk <Nie mam konta>.4. System wyświetla formularz rejestracji nowego konta do systemu.5. Użytkownik wprowadza dane takie jak: login, hasło oraz powtarza hasło.6. Użytkownik zatwierdza wpisane informacje przyciskiem <Zarejestruj>, lub klikając klawisz „ENTER”7. System tworzy konto w systemie, wyświetla komunikat o sukcesie, a następnie autoryzuje użytkownika i przenosi go do ekranu głównego aplikacji. <p><u>Przebiegi alternatywne:</u></p> <p>7a. System wyświetla informację o za krótkim hasle.</p> <p>7b. System wyświetla informację mówiącą, że pola „hasło” i „powtórz hasło” nie są identyczne.</p> <p>7c. System wyświetla informację o tym, że użytkownik o podanym loginie istnieje już w systemie.</p>
Efekt końcowy	Utworzenie konta w systemie oraz poprawna autoryzacja użytkownika.
Uwagi	Hasło powinno składać się z minimum 8 znaków.

3.3.4 Rozwinięcie opisu przypadku użycia „Zarządzaj kontem”.

Nowo powstałe konto w systemie zawsze na początku posiada startowe, domyślne wartości, takie jak wyświetlana nazwa użytkownika czy rola konta. Dzięki funkcjom realizowanym przy zarządzania swoim kontem można zmienić nazwę użytkownika, a administrator może przypisać komuś uprawnienia osoby obsługującej system.

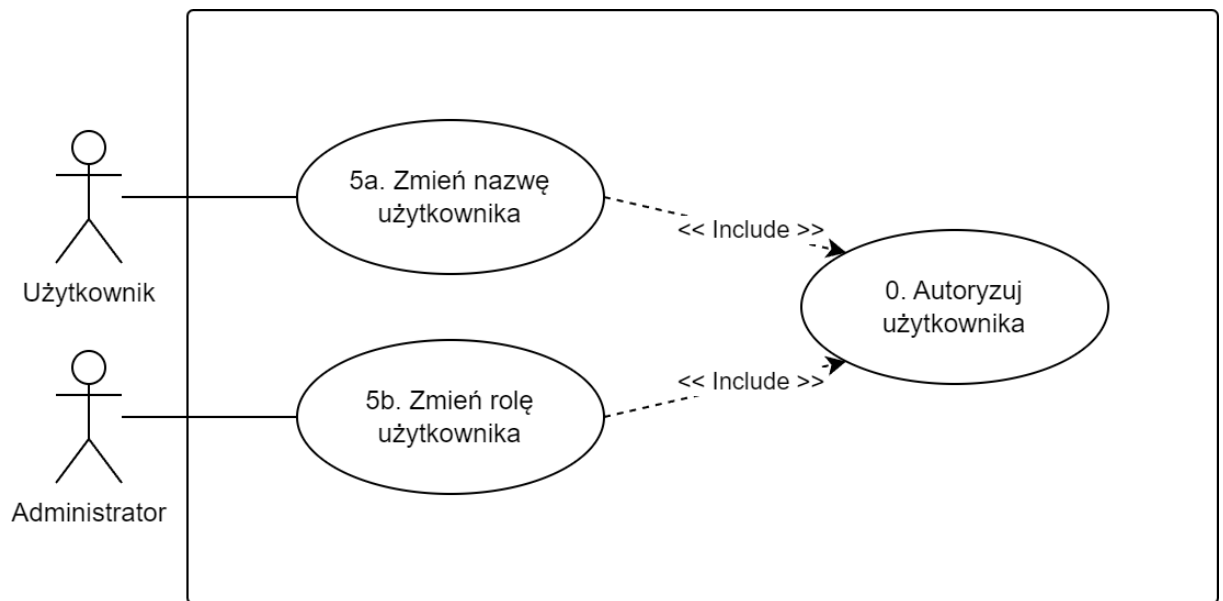


Diagram 4 Szczegółowy diagram przypadku użycia „Zarządzaj kontem”

Scenariusz przypadku użycia „Zmień nazwę użytkownika”

Tabela 3 Scenariusz przypadku użycia „Zmień nazwę użytkownika”

Identyfikator przypadku użycia	5a
Nazwa przypadku użycia	Zmień nazwę użytkownika
Cel przypadku użycia	Poprawna zmiana wyświetlanej, publicznej nazwy użytkownika.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none"> 1. Użytkownik wyraża chęć zmiany nazwy użytkownika klikając na pole <Cześć **aktualna nazwa** !>. 2. System wyświetla formularz zmiany nazwy w systemie. 3. Użytkownik wprowadza dane w postaci nazwy użytkownika. 4. Użytkownik zatwierdza nową nazwę użytkownika przyciskiem <✓>, lub klikając klawisz „ENTER”. 5. System pomyślnie wprowadza dane do bazy danych, a następnie wyświetla komunikat o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <p>3a.1 Użytkownik anuluje swoje działania klikając przycisk <X>.</p> <p>3a.2 System zamyka formularz.</p> <p>5a System zamyka formularz i wyświetla komunikat o braku wprowadzenia żadnych zmian.</p>
Efekt końcowy	Pomyślne ustawienie nazwy użytkownika w systemie.
Uwagi	brak

3.3.5 Rozwinięcie opisu przypadku użycia „Zarządzaj budżetami”.

Drugim najważniejszą funkcjonalnością systemu jest zarządzaniem Budżetami użytkownika. To dzięki tym funkcjom klient może tworzyć nowe Rachunki, w których zapisywać będzie swoje wydatki i przychody, a także dodawać do nich nowe osoby w celu tworzenia wspólnego miejsca do monitorowania przepływu pieniędzy w ciągu miesiąca.

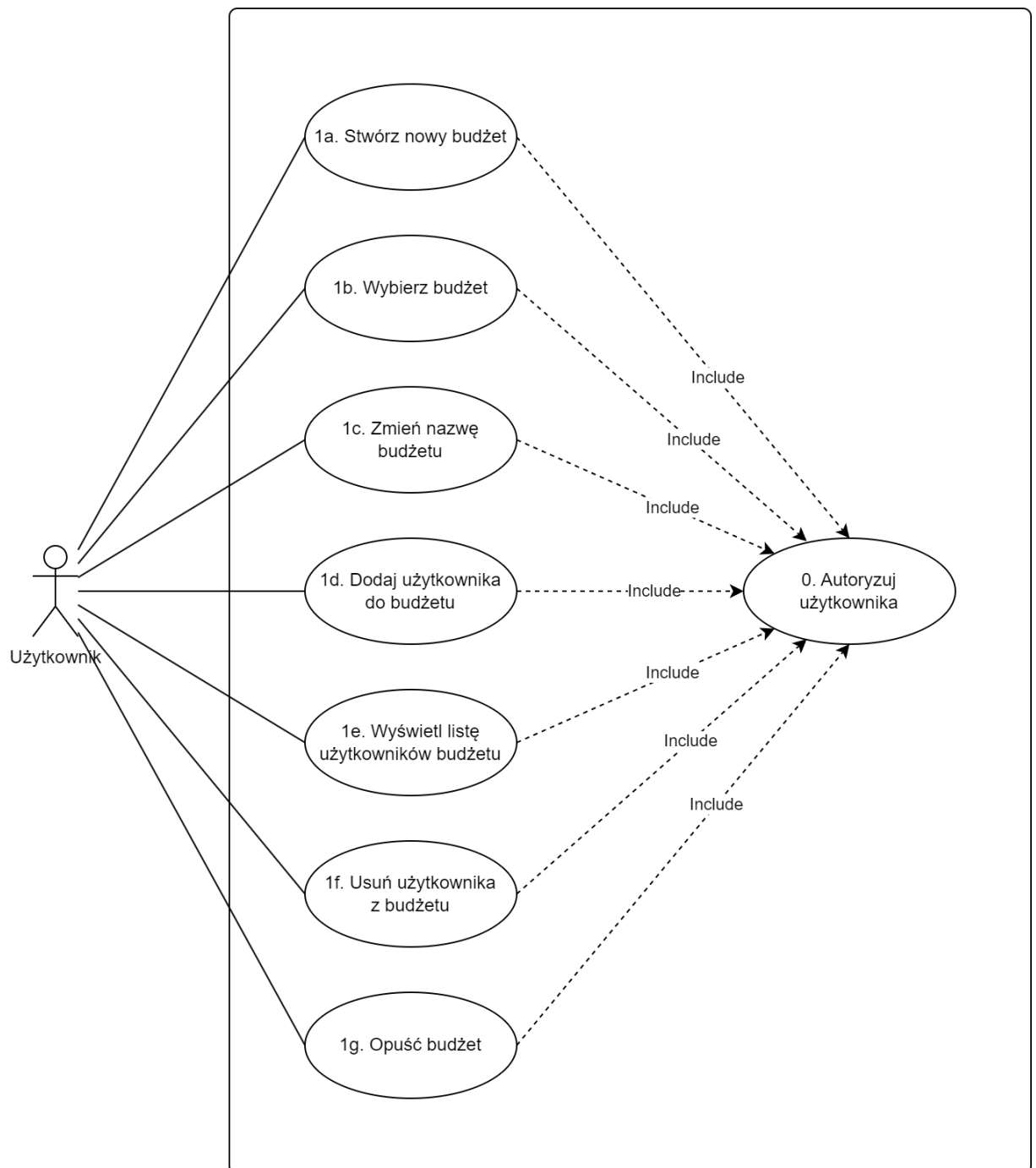


Diagram 5 Szczegółowy diagram przypadku użycia „Zarządzaj budżetami”

Scenariusz przypadku użycia „Stwórz nowy budżet”

Tabela 4 Scenariusz przypadku użycia „Stwórz nowy budżet”

Identyfikator przypadku użycia	1a
Nazwa przypadku użycia	Stwórz nowy budżet
Cel przypadku użycia	Wprowadzenie do systemu nowego budżetu przypisanego do konta użytkownika.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none"> 1. Użytkownik wyraża chęć utworzenia nowego budżetu poprzez naciśnięcie przycisku <+ Dodaj nowy budżet>. 2. System wyświetla formularz tworzenia nowego budżetu. 3. Użytkownik wprowadza dane w postaci nazwy budżetu. 4. Użytkownik zatwierdza formularz klikając przycisk <+>, lub klikając klawisz „ENTER”. 5. System poprawnie tworzy nowy budżet, a następnie przenosi użytkownika na ekran z listą transakcji. <p><u>Przebiegi alternatywne:</u></p> <p>0a. Użytkownik na urządzeniu mobilnym wysuwa prawy panel aplikacji klikając przycisk <≡>.</p> <p>3a.1. Użytkownik rezygnuje z chęci tworzenia nowego budżetu klikając przycisk <X>.</p> <p>3a.3. System zamyka formularz.</p> <p>5a. System wyświetla komunikat o braku wprowadzonej nazwy budżetu.</p>
Efekt końcowy	Pomyślne stworzenie budżetu i przypisanie go do konta klienta.
Uwagi	brak

Scenariusz przypadku użycia „Zmień nazwę budżetu”

Tabela 5 Scenariusz przypadku użycia „Zmień nazwę budżetu”

Identyfikator przypadku użycia	1c
Nazwa przypadku użycia	Zmień nazwę budżetu
Cel przypadku użycia	Poprawna zmiana nazwy budżetu
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none">1. Użytkownik wyraża chęć zmiany nazwy budżetu klikając na pole <*<aktualna nazwa budżetu*>, w panelu środkowym aplikacji.2. System wyświetla formularz zmiany nazwy w systemie.3. Użytkownik wprowadza dane w postaci nazwy budżetu.4. Użytkownik zatwierdza nową nazwę przyciskiem <✓>, lub klikając klawisz „ENTER”.5. System pomyślnie wprowadza dane do bazy danych, a następnie wyświetla komunikat o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <p>3a.1 Użytkownik anuluje swoje działania klikając przycisk <X>.</p> <p>3a.2 System zamyka formularz.</p> <p>5a System zamyka formularz i wyświetla komunikat o braku wprowadzenia żadnych zmian.</p>
Efekt końcowy	Pomyślna modyfikacja nazwy budżetu.
Uwagi	brak

Scenariusz przypadku użycia „Dodaj użytkownika do budżetu”

Tabela 6 Scenariusz przypadku użycia „Dodaj użytkownika do budżetu”

Identyfikator przypadku użycia	1d
Nazwa przypadku użycia	Dodaj użytkownika do budżetu
Cel przypadku użycia	Przypisanie nowego członka do budżetu.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie i wybranie budżetu.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none"> 1. Użytkownik otwiera okno z dodatkowymi działaniami, klikając przycisk <!>, po prawej stronie od nazwy budżetu w panelu środkowym. 2. System wyświetla okno przedstawiające dodatkowe działania wobec budżetu. 3. Użytkownik klika przycisk <Dodaj użytkownika>. 4. System wyświetla okno wraz z formularzem dodawania nowego członka do budżetu. 5. Użytkownik wprowadza ID użytkownika, którego chce dodać do listy osób w budżecie. 6. Użytkownik zatwierdza wprowadzone dane przyciskiem <Dodaj> , lub klikając klawisz „ENTER”. 7. System pomyślnie dodaje nową osobę do budżetu, a następnie wyświetla komunikat o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <ol style="list-style-type: none"> 5a.1. Użytkownik rezygnuje z chęci modyfikacji listy członków w budżecie klikając przycisk <X> 5a.2. System zamyka okno. 7a. System wyświetla komunikat informujący o błędnym numerze ID. 7b. System wyświetla komunikat o przynależności osoby o podanym ID w budżecie.
Efekt końcowy	Lista osób przypisanych do budżetu zwiększa się.
Uwagi	ID użytkownika, który ma zostać dodany do budżetu powinno zostać przekazane tradycyjnymi środkami komunikacji: werbalnie, mailowo lub telefonicznie.

Scenariusz przypadku użycia „Wyświetl listę użytkowników budżetu”

Tabela 7 Scenariusz przypadku użycia „Wyświetl listę użytkowników budżetu”

Identyfikator przypadku użycia	1e
Nazwa przypadku użycia	Wyświetl listę użytkowników budżetu
Cel przypadku użycia	Pokazanie wszystkich aktualnych użytkowników przypisanych do budżetu.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie.
Przebieg działań	<u>Przebieg podstawowy:</u> 1. Użytkownik otwiera okno z dodatkowymi działaniami, klikając przycisk < >, po prawej stronie od nazwy budżetu w panelu środkowym. 2. System wyświetla okno przedstawiające dodatkowe działania wobec budżetu. 3. Użytkownik klika przycisk „Wyświetl użytkowników”. 4. System wyświetla okno z aktualnymi użytkownikami w budżecie.
Efekt końcowy	Zaprezentowanie użytkownikowi aktualnej listy użytkowników w budżecie.
Uwagi	brak

Scenariusz przypadku użycia „Usuń użytkownika z budżetu”

Tabela 8 Scenariusz przypadku użycia „Usuń użytkownika z budżetu”

Identyfikator przypadku użycia	1f
Nazwa przypadku użycia	Usuń użytkownika z budżetu
Cel przypadku użycia	Pomniejszenie listy członków przypisanych do budżetu o konkretnego użytkownika.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none"> 1. Użytkownik otwiera okno z dodatkowymi działaniami, klikając przycisk < >, po prawej stronie od nazwy budżetu w panelu środkowym. 2. System wyświetla okno przedstawiające dodatkowe działania wobec budżetu. 3. Użytkownik klika przycisk „Wyświetl użytkowników”. 4. System wyświetla okno z aktualnymi użytkownikami w budżecie. 5. Użytkownik wyraża chęć usunięcia innego członka z budżetu klikając przycisk <X>. 6. System wyświetla komunikat upewniający się czy użytkownik chce usunąć daną osobę z budżetu. 7. Użytkownik potwierdza działanie klikając przycisk <OK>. 8. System blokuje dostęp do budżetu usuniętej osobie, a następnie wyświetla komunikat o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <p>7a. Użytkownik anuluje działania klikając dowolne miejsce obszaru aplikacji poza komunikatem.</p>
Efekt końcowy	Zmniejszenie listy użytkowników mających dostęp do budżetu.
Uwagi	brak

Scenariusz przypadku użycia „Opuść budżet”

Tabela 9 Scenariusz przypadku użycia „Opuść budżet”

Identyfikator przypadku użycia	1g
Nazwa przypadku użycia	Opuść budżet
Cel przypadku użycia	Wypisanie się z listy członków danego budżetu.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none"> 1. Użytkownik otwiera okno z dodatkowymi działaniami, klikając przycisk <!>, po prawej stronie od nazwy budżetu w panelu środkowym. 2. System wyświetla okno przedstawiające dodatkowe działania wobec budżetu. 3. Użytkownik klika przycisk <Opuść budżet>. 4. System wyświetla komunikat upewniający się, czy użytkownik na pewno chce opuścić budżet. 5. Użytkownik potwierdza działanie poprzez kliknięcie przycisku <OK>. 6. System blokuje dostęp użytkownikowi do budżetu, a następnie wyświetla komunikat o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <ol style="list-style-type: none"> 6a.1. Jeśli użytkownik jest jedynym członkiem w budżecie system wyświetla komunikat upewniający się, czy użytkownik chce usunąć budżet. 6a.2 Użytkownik potwierdza działanie poprzez kliknięcie przycisku <OK>. 6a.3. System usuwa budżet, a następnie wyświetla komunikat o sukcesie.
Efekt końcowy	Opuszczenie budżetu przez użytkownika.
Uwagi	brak

3.3.6 Rozwinięcie opisu przypadku użycia „Zarządzaj transakcjami”.

Główną funkcją systemu jest dodawanie nowych transakcji i zarządzanie transakcjami istniejącymi. Założeniem korzystania z aplikacji jest to, że przy nowym wydatku lub przychodzie pieniądza, klient opisuje odpowiednio taką transakcję, dodając tytuł, kwotę, kategorię oraz opcjonalnie opis: tak przygotowaną transakcję przypisuje do budżetu, który jego zdaniem jest odpowiednim miejscem na jej przechowywanie. W przypadku chęci edycji czy usunięcia, może on zarządzać postami stworzonymi przez siebie.

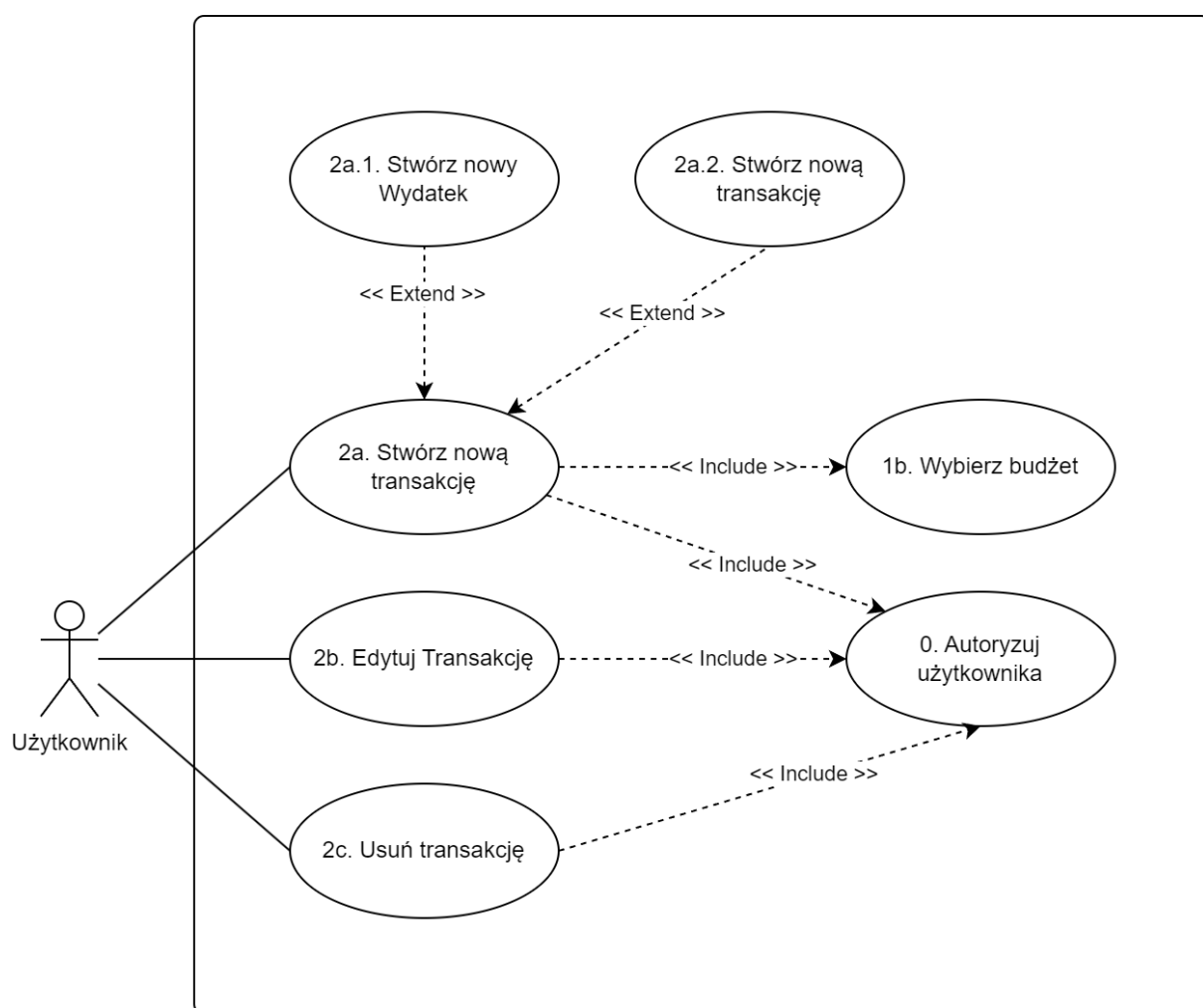


Diagram 6 Szczegółowy diagram przypadku użycia „Zarządzaj transakcjami”

Scenariusz przypadku użycia „Stwórz nową transakcję”

Tabela 10 Scenariusz przypadku użycia „Stwórz nową transakcję”

Identyfikator przypadku użycia	2a
Nazwa przypadku użycia	Stwórz nową transakcję
Cel przypadku użycia	Przypisanie nowego Wydatku lub Przychodu do budżetu.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie i wybranie konkretnego budżetu.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none">1. Użytkownik wyraża chęć stworzenia nowej transakcji klikając przycisk <Dodaj nową transakcję>.2. System wyświetla okno z formularzem.3. Użytkownik wprowadza niezbędne dane do stworzenia transakcji.4. Użytkownik zatwierdza wprowadzone informacje klikając przycisk <Dodaj transakcję>.5. System pomyślnie tworzy nową transakcję, a następnie wyświetla komunikat o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <p>3a. Użytkownik rezygnuje z chęci dodania transakcji klikając przycisk <X>, lub obszar aplikacji poza oknem.</p> <p>5a. System wyświetla komunikat informujący o konieczności podania tytułu.</p> <p>5b. System wyświetla komunikat informujący o konieczności podania kwoty transakcji.</p> <p>5c. System wyświetla komunikat informujący o konieczności wybrania kategorii.</p>
Efekt końcowy	Nowa transakcja pojawia się na liście operacji w budżecie.
Uwagi	brak

Scenariusz przypadku użycia „Edytuj transakcję”

Tabela 11 Scenariusz przypadku użycia „Edytuj transakcję”

Identyfikator przypadku użycia	2b
Nazwa przypadku użycia	Edytuj transakcję
Cel przypadku użycia	Zmiana danych w istniejącej transakcji.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie, wybranie konkretnego budżetu i wcześniejsze utworzenie transakcji.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none"> 1. Użytkownik wyraża chęć edycji poprzez kliknięcie przycisku <✎>, w prawym górnym rogu transakcji której treść chce zmienić. 2. System wyświetla okno z formularzem edycji transakcji. 3. Użytkownik wprowadza poprawione dane. 4. Użytkownik zatwierdza zmiany przyciskiem <Edytuj>. 5. System wprowadza edytuje dane w bazie danych, następnie wyświetla komunikat o sukcesie <p><u>Przebiegi alternatywne:</u></p> <ol style="list-style-type: none"> 3a. Użytkownik rezygnuje z chęci edycji transakcji klikając przycisk <X>, lub obszar aplikacji poza oknem. 5a. System wyświetla komunikat o konieczności podania tytułu transakcji. 5b. System wyświetla komunikat o konieczności podania kwoty operacji. 5c. System wyświetla komunikat o konieczności wprowadzenia daty.
Efekt końcowy	Pomyślna zmiana danych transakcji.
Uwagi	brak

Scenariusz przypadku użycia „Usuń transakcję”

Tabela 12 Scenariusz przypadku użycia „Usuń transakcję”

Identyfikator przypadku użycia	2c
Nazwa przypadku użycia	Usuń transakcję
Cel przypadku użycia	Usunięcie danej transakcji z budżetu.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie, wybranie konkretnego budżetu i wcześniejsze utworzenie transakcji.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none"> 1. Użytkownik wyraża chęć usunięcia poprzez kliknięcie przycisku <🗑️>, w prawym górnym rogu transakcji której chce się pozbyć. 2. System wyświetla komunikat upewniający się czy użytkownik na pewno chce usunąć transakcję. 3. Użytkownik potwierdza chęć swoich działań klikając przycisk <OK>. 4. System usuwa transakcję z bazy danych, a następnie wyświetla informację o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <p>3a. Użytkownik anuluje swoje działania klikając dowolne miejsce obszaru aplikacji poza komunikatem.</p>
Efekt końcowy	Transakcja została usunięta.
Uwagi	brak

3.3.7 Rozwinięcie opisu przypadku użycia „Zarządzaj notatkami”.

Zdarza się, że zachodzi potrzeba sporządzenia listy zakupów, zapisania ważnych dat lub listy zadań na dany okres. Takie i inne myśli użytkownik aplikacji jest w stanie przelać na wirtualny papier za pomocą funkcji realizowanych wobec niżej przedstawionego przypadku użycia. Klient znajdzie tutaj możliwość tworzenia nowych notatek, które są dostępne dla wszystkich członków budżetu, a także funkcję edycji i usuwania notatek istniejących w budżecie.

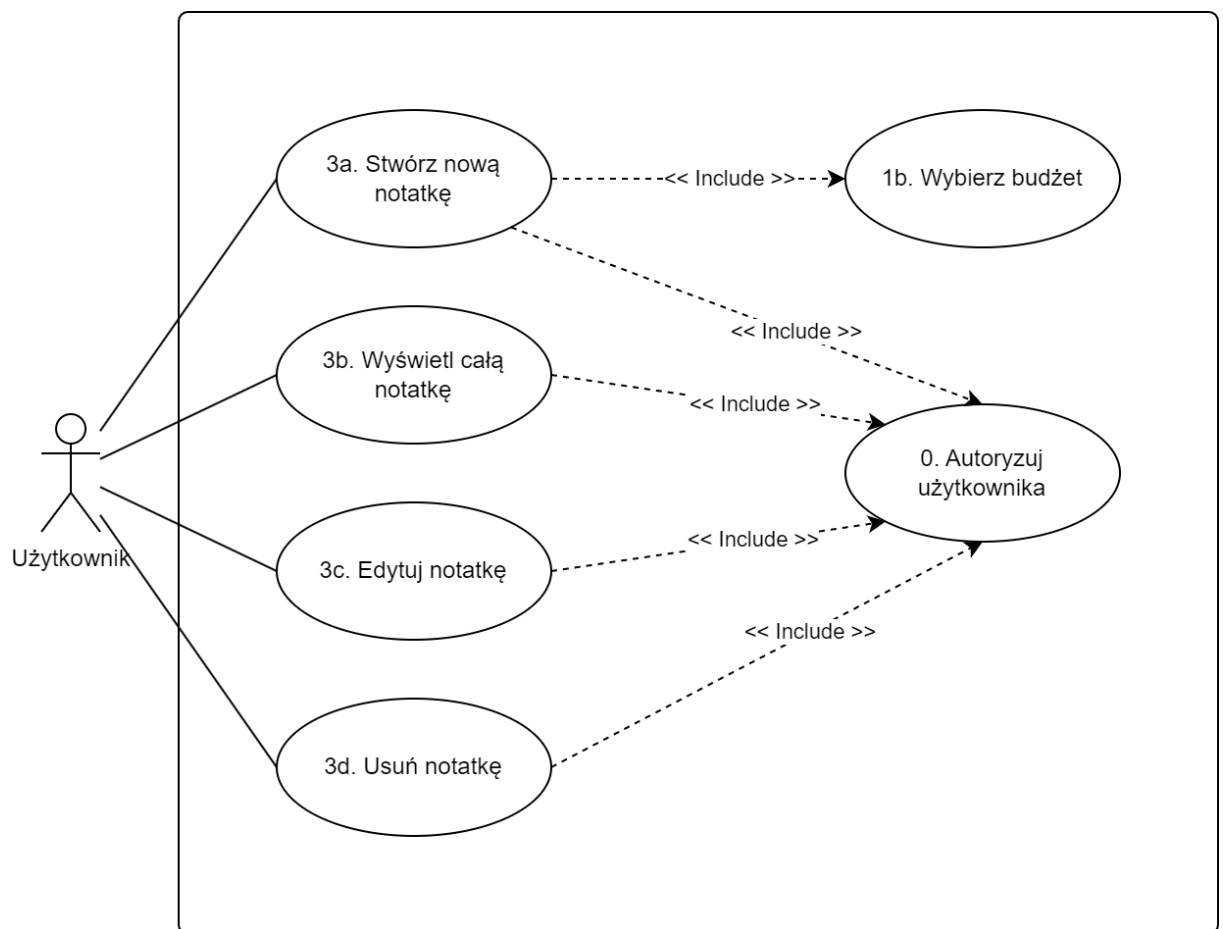


Diagram 7 Szczegółowy diagram przypadku użycia „Zarządzaj notatkami”

Scenariusz przypadku użycia „Stwórz nową notatkę”

Tabela 13 Scenariusz przypadku użycia „Stwórz nową notatkę”

Identyfikator przypadku użycia	3a
Nazwa przypadku użycia	Stwórz nową notatkę
Cel przypadku użycia	Dodanie nowej notatki do listy notatek w budżecie.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie i wybranie konkretnego budżetu.
Przebieg działań	<u>Przebieg podstawowy:</u> <ol style="list-style-type: none">1. Użytkownik wyraża chęć dodania nowej notatki do budżetu klikając przycisk <+ Dodaj notatkę>.2. System wyświetla okno z formularzem tworzenia notatki.3. Użytkownik wprowadza treść notatki w postaci tytułu i opisu.4. Użytkownik zatwierdza chęć utworzenia notatki klikając przycisk <Dodaj notatkę>.5. System pomyślnie tworzy nową notatkę, a następnie wyświetla komunikat o sukcesie.
Efekt końcowy	Nowa notatka pojawia się w liście notatek w lewym panelu aplikacji.
Uwagi	brak

Scenariusz przypadku użycia „Edytuj notatkę”

Tabela 14 Scenariusz przypadku użycia „Edytuj notatkę”

Identyfikator przypadku użycia	3c
Nazwa przypadku użycia	Edytuj notatkę
Cel przypadku użycia	Zmiana treści notatki.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie i wybranie konkretnego budżetu.
Przebieg działań	<p><u>Przebieg podstawowy:</u></p> <ol style="list-style-type: none">1. Użytkownik wyraża chęć zmiany treści notatki poprzez kliknięcie na nią na liście notatek.2. System wyświetla okno z formularzem edycji notatki.3. Użytkownik wprowadza treść notatki.4. Użytkownik zatwierdza chęć zapisania zmian treści notatki klikając przycisk <Zapisz>.5. System pomyślnie edytuje treść notatki, a następnie wyświetla komunikat o sukcesie. <p><u>Przebiegi alternatywne:</u></p> <p>5a. System zamyka okno i wyświetla komunikat o braku wprowadzeniu zmian.</p>
Efekt końcowy	Pomyślna zmiana treści notatki.
Uwagi	brak

Scenariusz przypadku użycia „Usuń notatkę”

Tabela 15 Scenariusz przypadku użycia „Usuń notatkę”

Identyfikator przypadku użycia	3d
Nazwa przypadku użycia	Usuń notatkę
Cel przypadku użycia	Usunięcie danej notatki z listy notatek budżetu.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie.
Przebieg działań	<u>Przebieg podstawowy:</u> 1. Użytkownik wyraża chęć usunięcia notatki poprzez kliknięcie na nią na liście notatek. 2. System wyświetla okno z formularzem edycji notatki. 3. Użytkownik zatwierdza chęć zapisania zmian treści notatki klikając przycisk <Usuń>. 5. System pomyślnie usuwa notatkę, a następnie wyświetla komunikat o sukcesie.
Efekt końcowy	Notatka zostaje usunięta z listy notatek w budżecie.
Uwagi	brak

3.3.8 Rozwinięcie opisu przypadku użycia „Utwórz raport miesięczny”.

Raport miesięczny jest świetnym przedstawieniem przeanalizowanych danych wszystkich transakcji w obrębie danego budżetu. Użytkownik za pomocą jednego prostego kroku, jakim jest wygenerowanie nowej analizy poprzez kliknięcie przycisku na stronie, może w jednym miejscu zobaczyć wszelkie potrzebne dane w formie czytelnych i przejrzystych wykresów dotyczących budżetu, do którego przynależy.

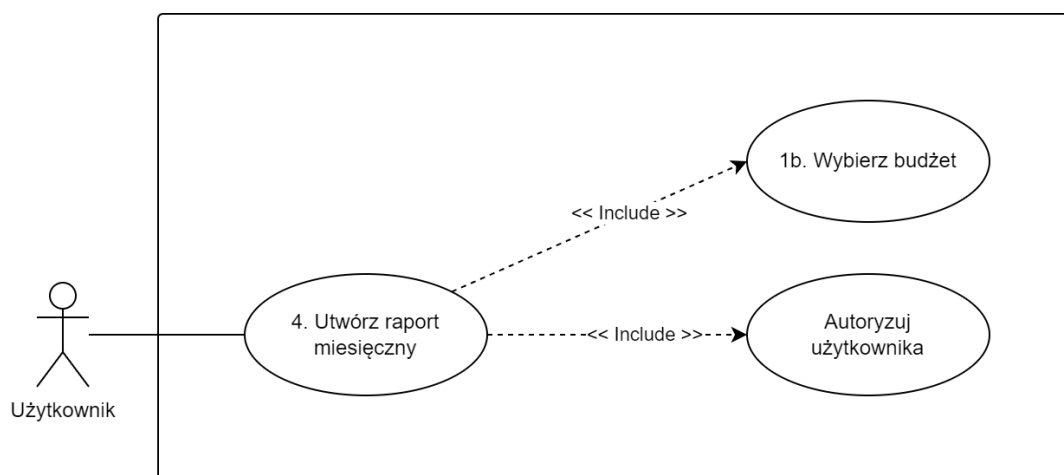


Diagram 8 Szczegółowy diagram przypadku użycia „Utwórz raport miesięczny”

Scenariusz przypadku użycia „Utwórz raport miesięczny”

Tabela 16 Scenariusz przypadku użycia „Utwórz raport miesięczny”

Identyfikator przypadku użycia	4
Nazwa przypadku użycia	Utwórz raport miesięczny
Cel przypadku użycia	Wyświetlenie podsumowania obrotu pieniądza w budżecie w danym miesiącu.
Aktorzy	Użytkownik
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika w systemie i wybranie konkretnego budżetu.
Przebieg działań	<u>Przebieg podstawowy:</u> 1. Użytkownik wyraża chęć wyświetlenia podsumowania transakcji klikając przycisk <Stwórz raport miesięczny>. 2. System wyświetla okno z najważniejszymi informacjami dotyczącymi budżetu.
Efekt końcowy	Pomyślne wyświetlenie raportu miesięcznego.
Uwagi	brak

3.3.9 Rozwinięcie opisu przypadku użycia „Zarządzaj kategoriami”.

Kategorie potrzebne są w celu dokładnego określania celu transakcji jaka miała miejsce, dzięki takim działaniom, w trakcie analizy system jest w stanie odpowiednio zamodelować i przedstawić raport miesięczny w którym uwzględni podział przychodów i wydatków z podziałem na kategorie.

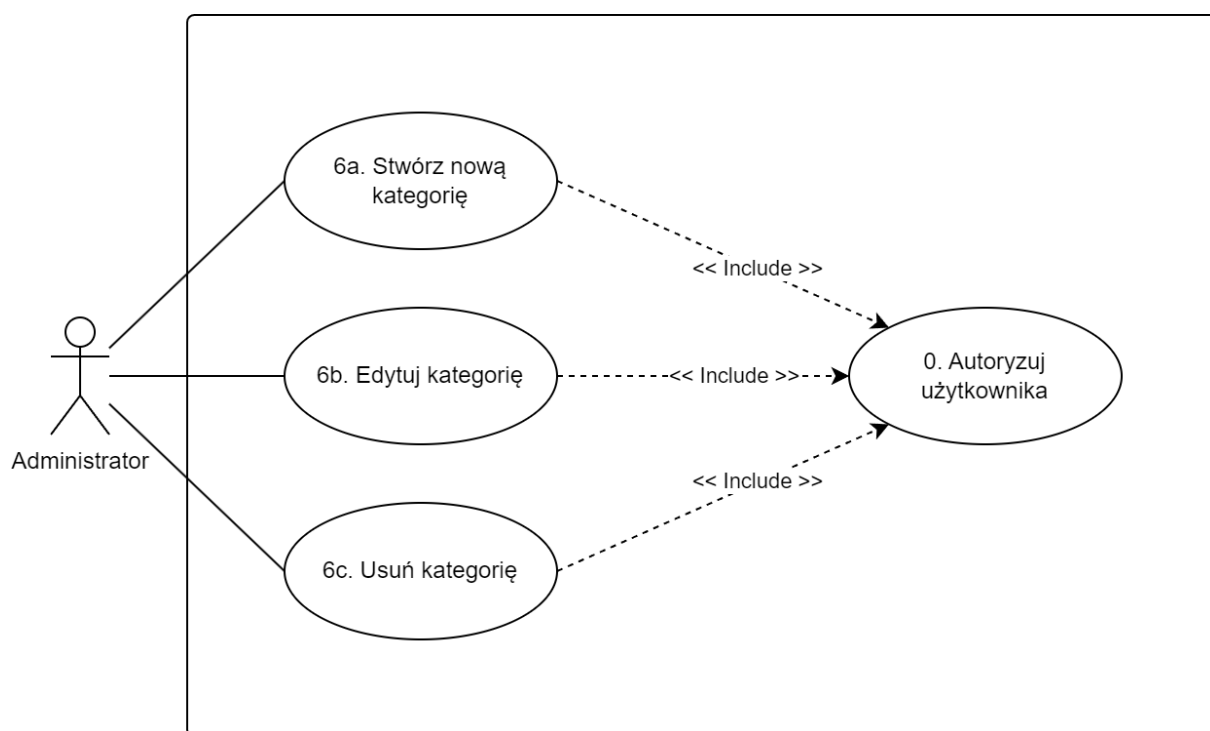


Diagram 9 Szczegółowy diagram przypadku użycia „Zarządzaj kategoriami”

Scenariusz przypadku użycia „Stwórz nową kategorię”

Tabela 17 Scenariusz przypadku użycia „Stwórz nową kategorię”

Identyfikator przypadku użycia	6a
Nazwa przypadku użycia	Stwórz nową kategorię
Cel przypadku użycia	Stworzenie nowej kategorii w systemie.
Aktorzy	Administrator
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika z prawami administratora w systemie.
Przebieg działań	<u>Przebieg podstawowy:</u> <ol style="list-style-type: none">1. Administrator wyraża chęć dodania nowej kategorii przechodząc do panelu dodawania nowej kategorii.2. System wyświetla formularz dodawania kategorii.3. Administrator wprowadza dane nowej kategorii.4. Administrator zatwierdza chęć dodania kategorii klikając przycisk <Execute>.5. System dodaje nową kategorię do bazy danych i zwraca informacje o sukcesie.
Efekt końcowy	Nowa kategoria została dodana.
Uwagi	brak

Scenariusz przypadku użycia „Edytuj kategorię”

Tabela 18 Scenariusz przypadku użycia „Edytuj kategorię”

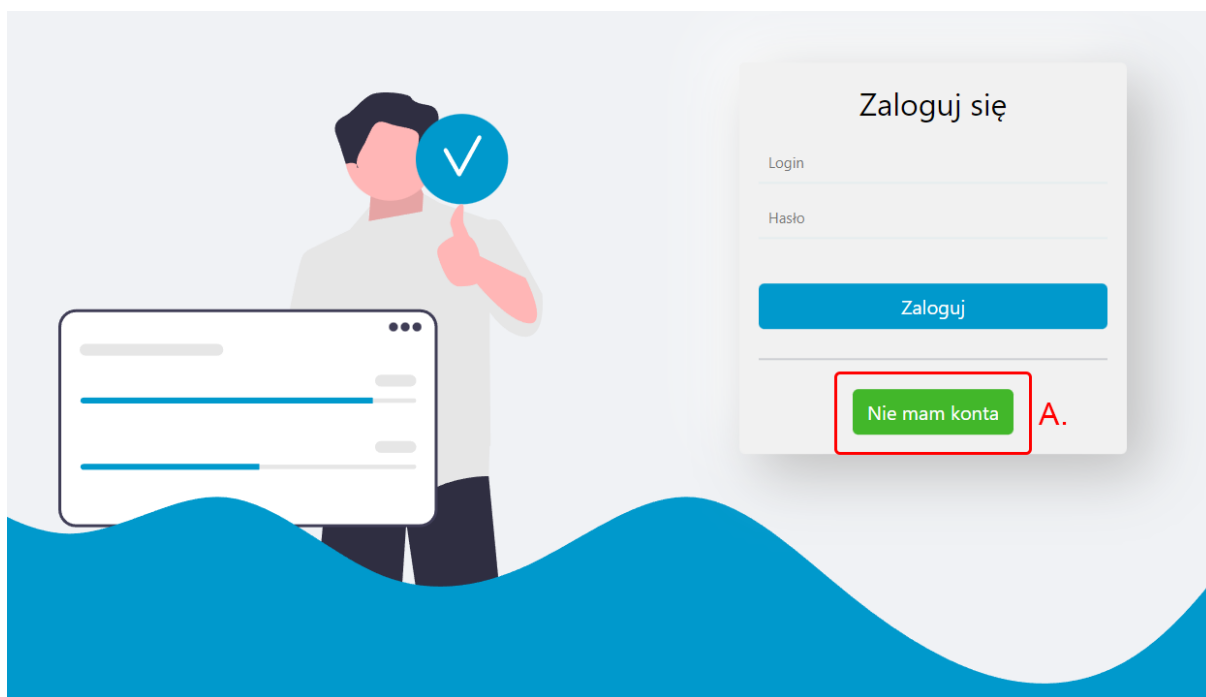
Identyfikator przypadku użycia	6b
Nazwa przypadku użycia	Edytuj kategorię
Cel przypadku użycia	Edycja danych istniejącej kategorii.
Aktorzy	Administrator
Warunek konieczny	System działa poprawnie i nie jest w trakcie prac konserwacyjnych.
Warunek wstępny	Pomyślne przejście procesu autoryzacji użytkownika z prawami administratora w systemie.
Przebieg działań	<u>Przebieg podstawowy:</u> <ol style="list-style-type: none">1. Administrator wyraża chęć dodania nowej kategorii przechodząc do panelu edytowania kategorii.2. System wyświetla formularz edycji kategorii.3. Administrator wprowadza dane kategorii.4. Administrator zatwierdza chęć dokonania zmian klikając przycisk <Execute>.5. System edytuje dane kategorii i zwraca informacje o sukcesie.
Efekt końcowy	Pomyślna edycja danych kategorii.
Uwagi	brak

4. Przewodnik po poszczególnych sekcjach i ekranach aplikacji

W poniższym rozdziale zaprezentowane zostaną zrzuty ekranów stworzonej aplikacji wraz z opisem każdej sekcji, która się w niej zawiera. Warstwa wizualna systemu została stworzona w dwóch motywach: jasnym i ciemnym. Dla przejrzystości dokumentu ekrany zostaną przedstawione z wykorzystaniem palety barw o jasnym charakterze. Motyw ciemny przedstawiony będzie na zrzucie ekranu głównego aplikacji.

4.1 Autoryzacja użytkownika

Po wybraniu przez użytkownika strony internetowej aplikacji, zostanie mu wyświetlony ekran logowania do systemu przedstawiony na *Rysunku 1*.



Rysunek 1 Ekran autoryzacji użytkownika

A. – Przycisk odpowiedzialny formie autoryzacji użytkownika (Zaloguj/Zarejestruj)

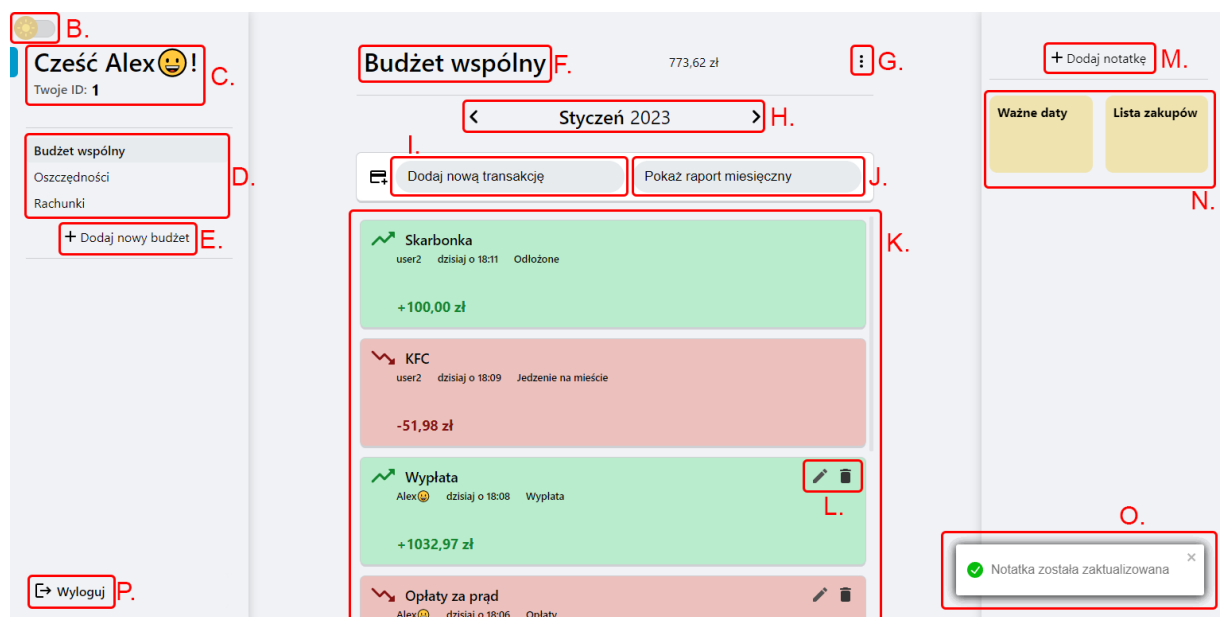
Następnie, aby poprawnie przejść proces autoryzacji, klient wprowadza swoje dane logowania do formularza przedstawionego na *Rysunku 2*:

The login form is titled "Zaloguj się". It features a text input field containing the username "aleksander". Below it is a password input field represented by a series of dots. A large blue button labeled "Zaloguj" is positioned below the password field. At the bottom of the form is a green button labeled "Nie mam konta".

Rysunek 2 Formularz logowania do systemu

4.2 Ekran główny aplikacji

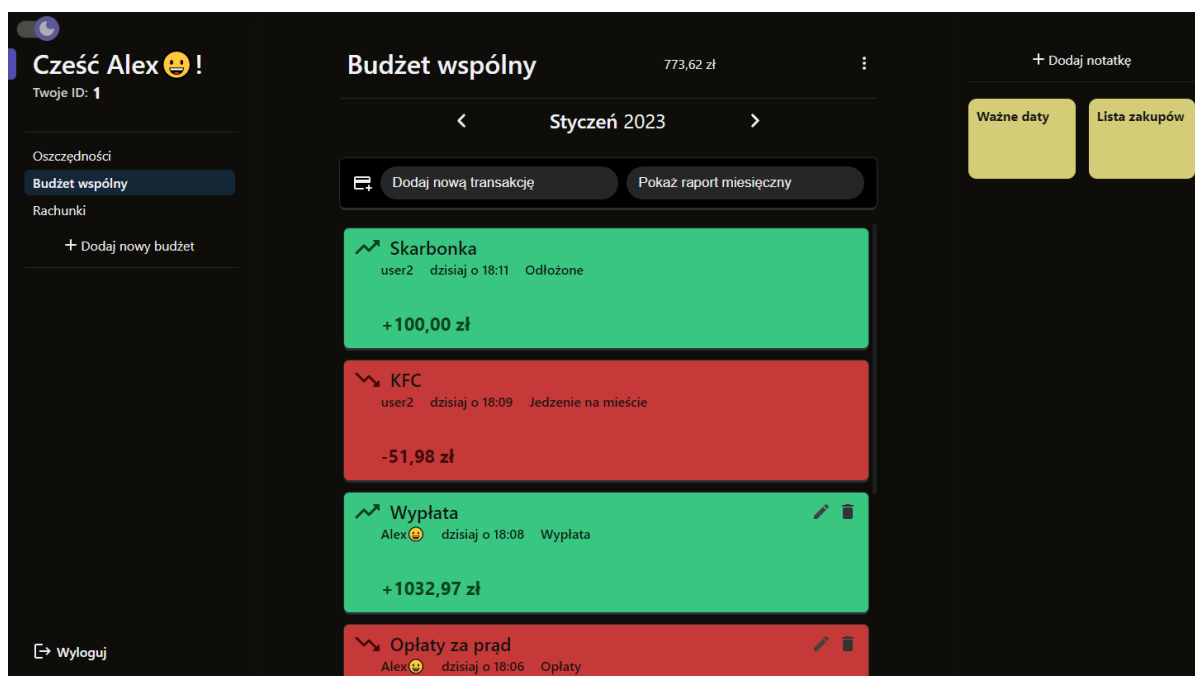
Autoryzowany użytkownik zostaje przeniesiony na ekran główny aplikacji widoczny na *Rysunku 3* w motywie jasnym oraz na *Rysunku 4* w ciemnej szacie graficznej. Poszczególne sekcje ekranu zostały opisane pod ilustracją.



Rysunek 3 Ekran główny aplikacji - motyw jasny

- B. Przycisk do zmiany motywu (jasny/ciemny)
- C. Dane użytkownika w formie wyświetlanej nazwy użytkownika i unikalnego numeru id
- D. Lista budżetów, których klient jest członkiem
- E. Przycisk odpowiedzialny za wyrażenie chęci dodania nowego budżetu
- F. Nazwa aktualnie wybranego budżetu
- G. Dodatkowe operacje w budżecie
- H. Miesiąc z jakiego użytkownik chce widzieć transakcje
- I. Przycisk odpowiedzialny za wyrażenie chęci dodania nowej transakcji
- J. Przycisk do generowania raportu miesięcznego z budżetu
- K. Lista transakcji w budżecie z konkretnego miesiąca
- L. Dodatkowe opcje modyfikacji konkretnej transakcji (edytuj/usuń)
- M. Przycisk odpowiedzialny za wyrażenie chęci dodania nowej notatki
- N. Lista notatek przypisanych do budżetu
- O. Komunikaty systemowe informujące o sukcesie wykonywanej akcji
- P. Przycisk do wylogowania się z aplikacji

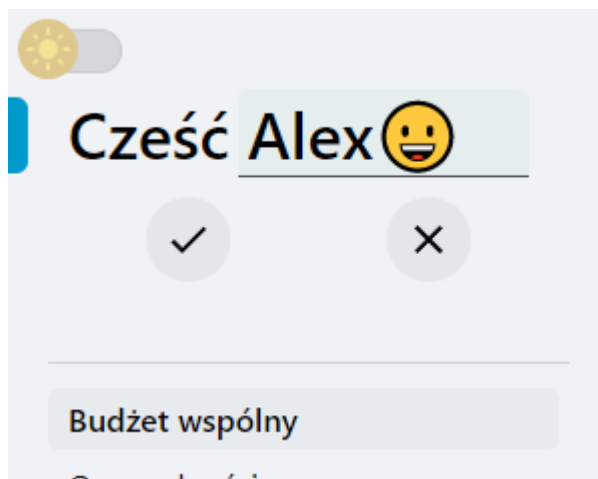
Użytkownik chcąc zmienić motyw aplikacji, powinien kliknąć przycisk oznaczony literą *B* na *Rysunku 3*:



Rysunek 4 Ekran główny aplikacji - motyw ciemny

4.3 Formularz zmiany nazwy użytkownika

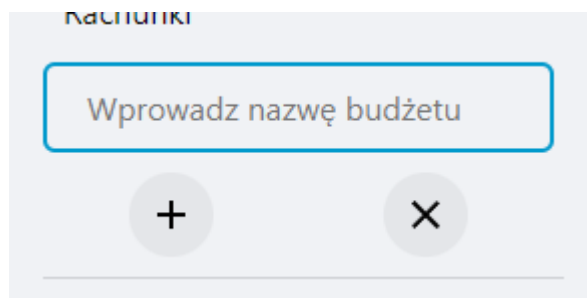
Osoba chcąc zmienić swoją wyświetlaną nazwę konta, powinna kliknąć na aktualną nazwę w sekcji C na Rysunku 3, a na ekranie pojawi się formularz widoczny na Rysunku 5, po zatwierdzeniu którego konto użytkownika zostanie zaktualizowane.

The screenshot shows a mobile application interface. At the top left, there is a yellow sun icon and a grey toggle switch. Below this, a blue vertical bar is partially visible. The main content area has a light blue background. It features a text input field containing 'Cześć Alex' followed by a yellow smiley face emoji. Below the input field are two circular buttons: one with a checkmark and one with an 'X'. At the bottom, there is a grey button labeled 'Budżet wspólny'.

Rysunek 5 Ekran z formularzem zmiany nazwy użytkownika

4.4 Formularz tworzenia nowego budżetu użytkownika

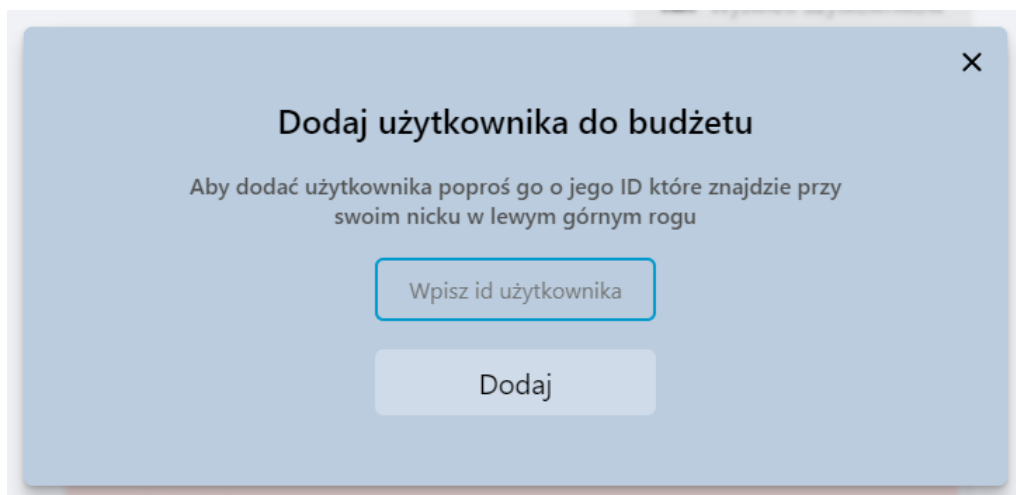
Klient, który wyraża chęć dodania nowego budżetu do swojego konta, proszony jest o kliknięcie sekcji E zaznaczonej na Rysunku 3. Po tej akcji system wyświetli formularz, który można zaobserwować na Rysunku 6. Wypełniwszy pole danych i zatwierdziwszy swoje działania, aplikacja pomyślnie utworzy nowy budżet i przypisze go do konta użytkownika.

The screenshot shows a mobile application interface. At the top, the word 'RACHUNKI' is visible. Below it, there is a text input field with the placeholder text 'Wprowadz nazwę budżetu'. Below the input field are two circular buttons: one with a plus sign and one with an 'X'. At the bottom, there is a grey button labeled 'Budżet wspólny'.

Rysunek 6 Ekran z formularzem tworzenia nowego budżetu

4.5 Formularz dodawania użytkownika do budżetu

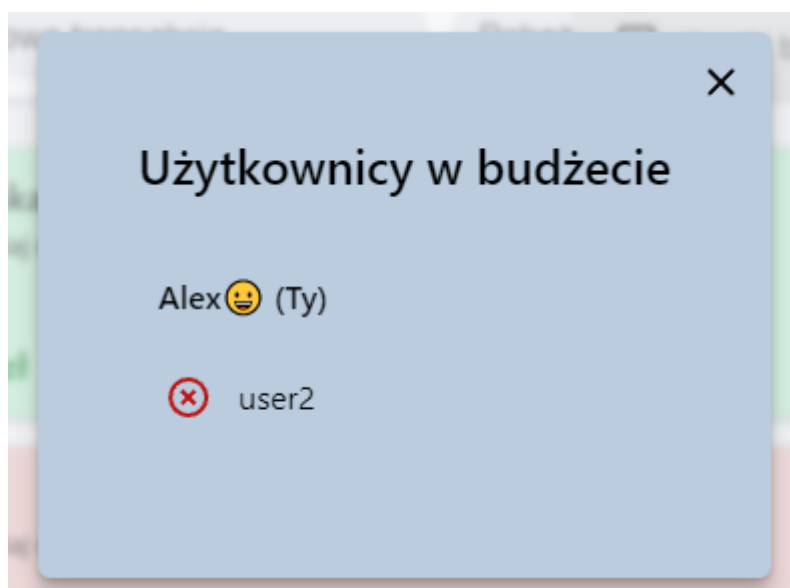
Chcąc przypisać nowych członków do swojego rachunku, użytkownik proszony jest o kliknięcie przycisku w sekcji *G* na ekranie na *Rysunku 3* a następnie wybrania odpowiedniej opcji. Aplikacja wyświetli okno z formularzem widoczne na *Rysunku 7*.



Rysunek 7 Okno wraz z formularzem dodawania nowego członka do budżetu

4.6 Lista użytkowników budżetu

Chcąc wyświetlić pełną listę członków budżetu, osoba powinna kliknąć przycisk w sekcji *G* na *Rysunku 3*, a następnie wybrać odpowiednią opcję. System wyświetli listę osób widoczną na *Rysunku 8*. Przy każdym z użytkowników na liście znajduje się przycisk <X>, odpowiedzialny za usuwanie osoby z budżetu, po kliknięciu którego system wyświetli zapytanie, czy na pewno chcemy wykonać tę akcję.



Rysunek 8 Okno z listą użytkowników w budżecie

4.7 Tworzenie nowej transakcji

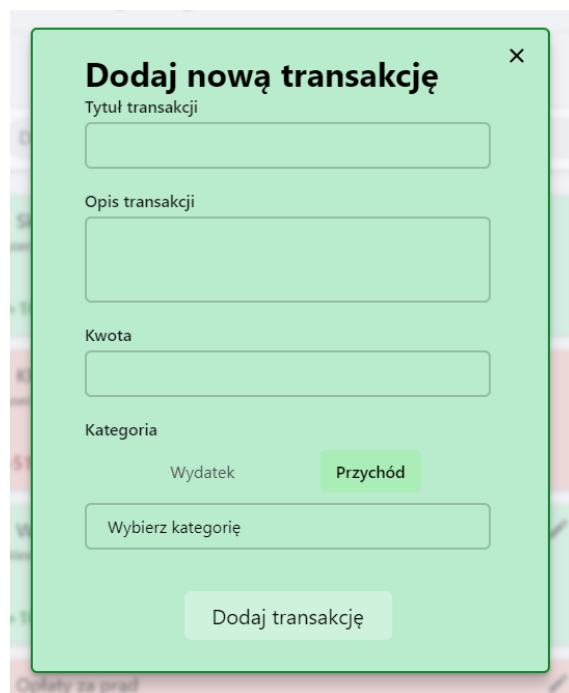
Po kliknięciu przycisku oznaczonego literką *I* na *Rysunku 3*, użytkownikowi zostaje wyświetlone okno z formularzem tworzenia nowego wydatku przedstawiony na *Rysunku 9*. Po zmianie charakteru transakcji z Wydatku na Przychód okno zmieni swój wygląd. Nową szatę graficzną prezentuje *Rysunek 10*.



Formularz dodawania nowego wydatku, o czerwonym tle i granicy. Tytuł formularza to "Dodaj nową transakcję" z ikoną zamknięcia (X) w prawym górnym rogu. Formularz zawiera następujące pola i elementy:

- Tytuł transakcji: pole tekstowe.
- Opis transakcji: pole tekstowe.
- Kwota: pole tekstowe.
- Kategoria: sekcja z dwoma przyciskami: "Wydatek" (aktywny, czerwony) i "Przychód" (nieaktywny, szary).
- Wybierz kategorię: pole tekstowe.
- Dodaj transakcję: przycisk na dole.

Rysunek 9 Formularz dodawania nowego wydatku



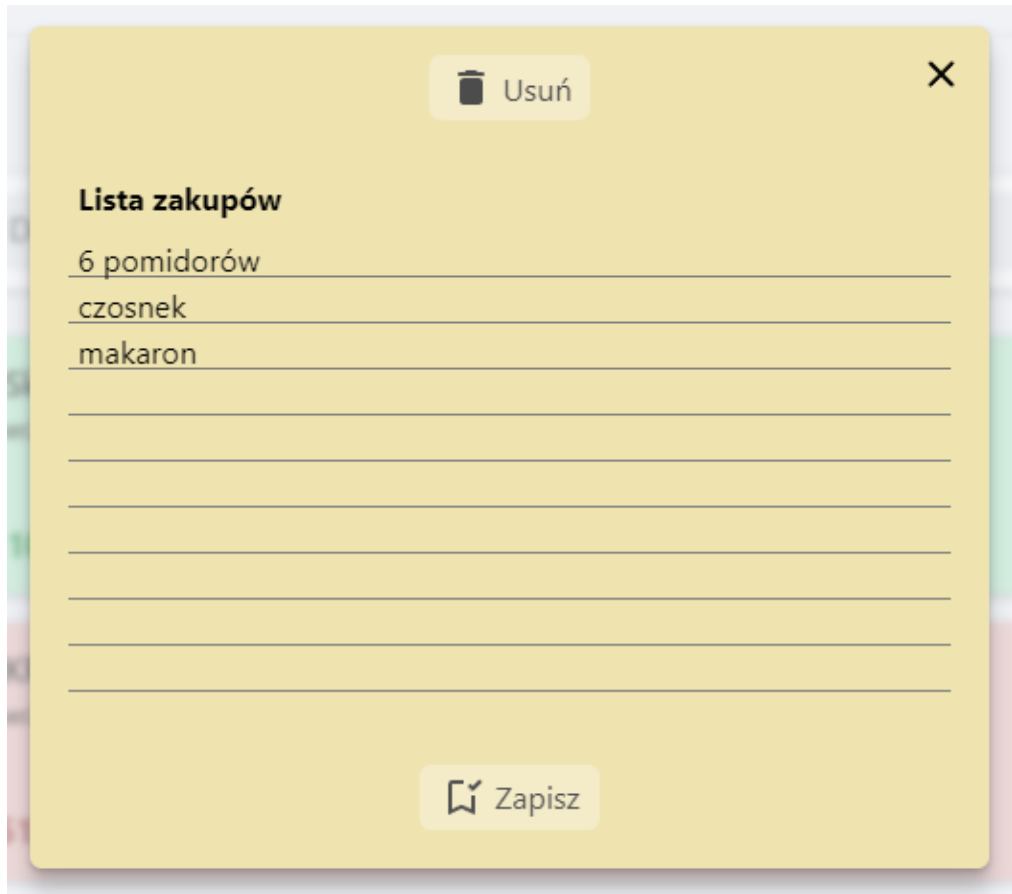
Formularz dodawania nowego przychodu, o zielonym tle i granicy. Tytuł formularza to "Dodaj nową transakcję" z ikoną zamknięcia (X) w prawym górnym rogu. Formularz zawiera następujące pola i elementy:

- Tytuł transakcji: pole tekstowe.
- Opis transakcji: pole tekstowe.
- Kwota: pole tekstowe.
- Kategoria: sekcja z dwoma przyciskami: "Wydatek" (nieaktywny, szary) i "Przychód" (aktywny, zielony).
- Wybierz kategorię: pole tekstowe.
- Dodaj transakcję: przycisk na dole.

Rysunek 10 Formularz dodawania nowego przychodu

4.8 Widok pojedynczej notatki

Kliknąwszy pojedynczy obiekt z listy notatek, oznaczonej literą *N* na *Rysunku 3*, system wyświetla okno z formularzem umożliwiającym modyfikację danych w niej zawartych. Notatka przedstawiona jest na *Rysunku 11*.



Lista zakupów

6 pomidorów

czosnek

makaron

Zapisz

Rysunek 11 Okno wraz z treścią notatki

4.9 Edycja istniejącej transakcji

Użytkownik chcąc edytować wcześniej stworzoną przez niego transakcję, powinien kliknąć odpowiedni przycisk w sekcji oznaczonej literką *L* na *Rysunku 3*. Po tej akcji system wyświetli okno wraz z formularzem, w którym klient może manipulować danymi, a po zatwierdzeniu działania aplikacja wprowadzi nową treść do bazy danych.

Edytuj transakcję ✕

Tytuł transakcji

Wypłata

Opis transakcji

Kwota (zł)

1032,97

Data transakcji

15.01.2023 18:08:04 📅

Kategoria

Wydatek ☐ Przychód ☒

Wypłata ▼

Edytuj

Rysunek 12 Formularz edycji istniejącej transakcji

5. Testowanie aplikacji

Testowanie powstałego systemu jest bardzo ważnym krokiem przed wdrożeniem go do sieci. Fundamentalnym założeniem aplikacji, która jest częścią pracy inżynierskiej, było zaimplementowanie intuicyjnego i zachęcającego do korzystania oprogramowania. Testy jakie zostały przeprowadzone na gotowym systemie, miały formę prostych zadań, a realizowane były przez osoby, które nigdy dotąd z aplikacji nie korzystały. Zespół testerów składał się z 5 osób w przedziale wiekowym 20-60 lat o różnym stopniu wykształcenia i umiejętnościach obsługi komputera. Najważniejszym aspektem doboru osób testujących było wybranie tych, którzy wyrazili chęć korzystania z aplikacji o takim zastosowaniu.

Każda osoba uczestnicząca w wydarzeniu miała do rozwiązania 5 zadań, a po skończeniu wszystkich została poproszona o wyrażenie swojej opinii na temat funkcjonowania systemu.

5.1 Zadania testujące

Poniżej zaprezentowane zostały zadania jakie testerzy dostali do rozwiązania, krótki opis i uwagi, jakie zgłosiły osoby testujące.

5.1.1 Zadanie nr 1

Stwórz nowe konto w systemie.

Użytkownicy nie mieli podyktowanych żadnych wytycznych odnośnie danych, jakie muszą wprowadzić, mogli wpisać bardzo długie albo – przeciwnie – bardzo krótkie hasło. Każdy z testerów przeszedł zadanie bez większych trudności i intuicyjnie wprowadził hasło składające się minimum z 8 znaków

5.1.2 Zadanie nr 2

Stwórz nowy budżet, nadaj mu nazwę, a następnie dodaj do niego użytkownika o numerze id 8

Testerzy po zalogowaniu się do systemu jednogłośnie stwierdzili, że strona prezentuje się nadzwyczaj przejrzysto: jest uporządkowana, a wszystkie funkcje przedstawione w sposób klarowny. Bez problemu odnaleźli przycisk do dodania nowego budżetu i przypisali nowy obiekt do swojego konta. Osoby młodsze nie miały problemu z rozwiązaniem zadania do końca, natomiast testerzy, którzy wychowali się w czasach początków cyfryzacji życia codziennego, potrzebowali pomocy w odnalezieniu dodatkowych operacji na budżecie.

5.1.3 Zadanie nr 3

Wybierasz się na zakupy, sporządź listę produktów jakie chcesz kupić w sklepie.

Osoby testujące od razu odnalazły odpowiednie przyciski odpowiedzialne za manipulowanie notatkami w budżecie. Każdy wprowadził dane, które jego zdaniem powinny zostać zapisane w treści i tytule, a następnie zatwierdził je, tworząc tym samym nowy obiekt w systemie

5.1.4 Zadanie nr 4

Byłaś/Byłeś dzisiaj na zakupach i kupiłaś/kupiłeś rzeczy do przygotowania obiadu i kolacji. Udokumentuj to zdarzenie w budżecie.

Jednogłośnie opinią, jaką wyrazili wszyscy uczestnicy, było stwierdzenie, że skoro aplikacja jest przeznaczona do monitorowania swoich wydatków, to jej atutem jest duży i dobrze widoczny przycisk odpowiedzialny za dodawanie nowej transakcji. Bez wahania wybrali właściwą opcję i wprowadzili nowe wydatki do systemu.

5.1.5 Zadanie nr 5

Spróbuj na co dzień korzystać z aplikacji. Klikaj, próbuj, baw się tym oprogramowaniem. Zadanie wykonuj przez 7 dni, a swoje odczucia pozytywne i negatywne zapisuj.

Jest to zadanie, które początkowo sprawiało najwięcej kłopotów i wywołało mieszane uczucia, jednak obawa przed korzystaniem z aplikacji minęła niedługo później — dwóch z testerów już kilka godzin od przedstawienia tego zadania wyraziło pozytywną opinię na temat aplikacji. Po pełnych 7 dniach, gdy autor ponownie spotkał się z testerami, przedstawili oni swoje zadowolenie i zasygnalizowali istotny plus aplikacji: wygodę, jaką płynie z korzystania z systemu.

5.2 Wnioski płynące z testowania aplikacji

Stworzona aplikacja, tak jak zakładano, okazała się intuicyjna dla użytkownika, a osoby chętne do korzystania z niej, wskazywały satysfakcję z funkcjonalności, jakie zostały zaimplementowane. Testerzy zgodnie podkreślili istotę funkcji generowania miesięcznych podsumowań w postaci raportu, w którym dane transakcji przedstawione są za pomocą czytelnych wykresów. Kolejnym, zdaniem osób testujących, atutem jest możliwość prowadzenia budżetu przez kilka osób jednocześnie, usprawnia to kontrolowanie wydatków w jednym gospodarstwie domowym przez wszystkich użytkowników.

Testerzy zostali także poproszeni o wystawienie opinii w postaci punktów w skali od 0 do 10. Uśrednioną oceną, jaką otrzymała aplikacja jest 9,66. Tak wysoka ocena pokazuje, że aplikacja została dobrze zaprojektowana i stworzona do potrzeb typowego użytkownika.

6. Podsumowanie i wnioski

Główny cel niniejszej pracy inżynierskiej, jakim było stworzenie aplikacji webowej umożliwiającej zarządzanie i monitorowanie wydatków w budżecie domowym, został wykonany zgodnie z założeniami projektowymi. Gotowy produkt zgadza się z dokładnie przeanalizowanymi funkcjonalnościami, jakie powinien spełniać system o takim zastosowaniu. Podczas implementacji nie pojawiały się znaczne problemy. Z wdrożonego systemu, przy użyciu darmowych hostingów takich jak „render.com” i „netlify.app”, można korzystać pod adresem: <https://homeorganizer.netlify.app/>.

Podczas korzystania z aplikacji trzeba mieć jednak na uwadze charakter i politykę darmowych serwisów, na których system został wdrożony. Kiedy program odpowiedzialny za obsługę zapytań serwerowych wejdzie w stan uśpienia, jego ponowne uruchomienie za pomocą pierwszego zapytania może potrwać nawet do 30 sekund, jednak każde kolejne będzie obsługiwane szybko i sprawnie.

Bibliografia

- [1] Martin, Robert C., and Paweł Gonera. „Czysty Kod: podręcznik Dobrego Programisty”. Helion, 2010
- [2] Susan Weinschenk. „100 rzeczy, które każdy projektant powinien wiedzieć o potencjalnych klientach”. Helion, 2013
- [3] Joe Morgan. „How To Code in React.js”. DigitalOcean 2021
- [4] Tiangolo. „FastAPI docs”, <https://fastapi.tiangolo.com/>
- [5] Meta Platforms, Inc. „ReactJS docs”, <https://reactjs.org/docs/getting-started.html>, 2023
- [6] Chart.js. „Chart.js docs”, <https://www.chartjs.org/docs/latest/>, 2023
- [7] SweetAlert. „SweetAlert2 docs”, <https://sweetalert2.github.io/>
- [8] “ JavaScript.” Wikipedia, 18 January 2023, en.wikipedia.org/wiki/JavaScript
- [9] “ Python (programming language).” Wikipedia, 17 January. 2023, [en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))
- [10] “ JSON Web Token” Wikipedia, 25 December 2022, https://en.wikipedia.org/wiki/JSON_Web_Token

Spis obrazów

Rysunek 1 Ekran autoryzacji użytkownika	40
Rysunek 2 Formularz logowania do systemu	41
Rysunek 3 Ekran główny aplikacji - motyw jasny	41
Rysunek 4 Ekran główny aplikacji - motyw ciemny	42
Rysunek 5 Ekran z formularzem zmiany nazwy użytkownika.....	43
Rysunek 6 Ekran z formularzem tworzenia nowego budżetu	43
Rysunek 7 Okno wraz z formularzem dodawania nowego członka do budżetu	44
Rysunek 8 Okno z listą użytkowników w budżecie	44
Rysunek 9 Formularz dodawania nowego wydatku	45
Rysunek 10 Formularz dodawania nowego przychodu	45
Rysunek 11 Okno wraz z treścią notatki	46
Rysunek 12 Formularz edycji istniejącej transakcji	47

Spis tabel

Tabela 1 Scenariusz przypadku użycia „Zaloguj się na istniejące konto”	17
Tabela 2 Scenariusz przypadku użycia „Stwórz nowe konto użytkownika”	18
Tabela 3 Scenariusz przypadku użycia „Zmień nazwę użytkownika”	20
Tabela 4 Scenariusz przypadku użycia „Stwórz nowy budżet”	22
Tabela 5 Scenariusz przypadku użycia „Zmień nazwę budżetu”	23
Tabela 6 Scenariusz przypadku użycia „Dodaj użytkownika do budżetu”	24
Tabela 7 Scenariusz przypadku użycia „Wyświetl listę użytkowników budżetu”	25
Tabela 8 Scenariusz przypadku użycia „Usuń użytkownika z budżetu”	26
Tabela 9 Scenariusz przypadku użycia „Opuść budżet”	27
Tabela 10 Scenariusz przypadku użycia „Stwórz nową transakcję”	29
Tabela 11 Scenariusz przypadku użycia „Edytuj transakcję”	30
Tabela 12 Scenariusz przypadku użycia „Usuń transakcję”	31
Tabela 13 Scenariusz przypadku użycia „Stwórz nową notatkę”	33
Tabela 14 Scenariusz przypadku użycia „Edytuj notatkę”	34
Tabela 15 Scenariusz przypadku użycia „Usuń notatkę”	35
Tabela 16 Scenariusz przypadku użycia „Stwórz raport miesięczny”	36
Tabela 17 Scenariusz przypadku użycia „Stwórz nową kategorię”	38
Tabela 18 Scenariusz przypadku użycia „Edytuj kategorię”	39

Spis diagramów

Diagram 1 Schemat bazy danych.....	12
Diagram 2 Główny diagram przypadków użycia	14
Diagram 3 Szczegółowy diagram przypadku użycia „Autoryzuj użytkownika”	16
Diagram 4 Szczegółowy diagram przypadku użycia „Zarządzaj kontem”	19
Diagram 5 Szczegółowy diagram przypadku użycia „Zarządzaj budżetami”	21
Diagram 6 Szczegółowy diagram przypadku użycia „Zarządzaj transakcjami”	28
Diagram 7 Szczegółowy diagram przypadku użycia „Zarządzaj notatkami”	32
Diagram 8 Szczegółowy diagram przypadku użycia „Stwórz raport miesięczny”	36
Diagram 9 Szczegółowy diagram przypadku użycia „Zarządzaj kategoriami”	37