

Platformy Programistyczne .NET i Java	
Kierunek, nazwa wydziału <i>Automatyka i Robotyka, ART</i> <i>Wydział Elektroniki, Fotoniki i Mikrosystemów</i>	Grupa projektowa, termin zajęć <i>Grupa 4, Środa 15¹⁵ – 16⁴⁵</i>
Imię, nazwisko, numer albumu <i>Kamil Orłowski, 259299</i> <i>Jędrzej Poniatowski, 259297</i>	Data oddania <i>07 maja 2023</i>
Tytuł <i>Aplikacja do obsługi długów</i>	Prowadzący <i>mgr Aneta Górniak</i>



SPRAWOZDANIE

Spis treści

1	ZAŁOŻENIA APLIKACJI	2
1.1	ZASTOSOWANIE APLIKACJI	2
1.2	OPIS FUNKCJONALNOŚCI	2
1.3	WYKORZYSTANE TECHNOLOGIE	2
2	PROCES TWORZENIA APLIKACJI	2
2.1	OPIS BAZY DANYCH	2
2.2	OGRANICZENIA	3
3	MOŻLIWOŚCI ROZWOJU APLIKACJI	3

1 Założenia aplikacji

Założeniem projektowym jest stworzenie aplikacji okienkowej w technologii .NET, działającej w systemie Windows i obsługującej podstawowe operacje na danych. W tym celu obmyślono rozwiązanie, które pomogłoby w zarządzaniu pożyczonymi pieniędzmi lub w rozszerzonej formie całym funduszem.

1.1 Zastosowanie aplikacji

Aplikacja umożliwia przechowywanie informacji o pożyczonych pieniądzach, dłużnikach oraz własnym portfelu. Umożliwione są operacje na podanym zbiorze danych w zakresie wymaganym dla łatwego użytkowania aplikacji. Pożyczki mogą być zarówno dane komuś lub wzięte od kogoś, w takim sensie są one wartością ujemną. Informacje o kontaktach są w pełni dostępne danemu użytkownikowi, który samemu może dodawać i usuwać dane kontakty. Oprócz bazowych funkcjonalności udostępnia się również sposoby na symulację i spłatę zobowiązań oraz powiadamianie kontaktów za pomocą maila.

1.2 Opis funkcjonalności

Dla rozwiązania problemu związanego ze stałym przechowywaniem danych jest zastosowanie lokalnej bazy danych, z którą program będzie się porozumiewał w celu wymiany informacji. Aplikacja ma formę okienkową, która wyświetla użytkownikowi dane pobrane z bazy danych oraz udostępnia ich modyfikację poprzez zestaw przycisków i okienek z formularzami. Funkcja wysyłania powiadomień osobom z kontaktów jest zapewniona poprzez zewnętrzne API. Pozostałe funkcjonalności są zapewnione bezpośrednio ze strony programu i języka, w którym został napisany.

1.3 Wykorzystane technologie

- Microsoft Visual Studio 2022 - środowisko programistyczne
- C# - język programowania
- .NET Framework 4.7.2 - platforma programistyczna
- Microsoft Entity Framework - baza danych
- Windows Presentation Foundation(WPF) - tworzenie okienek za pomocą xaml
- GitHub - repozytorium i kontrola rozwoju programu
- Dokumentacja w języku xaml stworzona automatycznie przez środowisko programistyczne VS
- Sandcastle - wykorzystany do stworzenia dokumentacji technicznej w formie strony internetowej

2 Proces tworzenia aplikacji

2.1 Opis bazy danych

Za pomocą narzędzia Entity Framework do mapowania obiektowo-relacyjnego stworzono lokalną bazę danych NoSQL składającą się z czterech encji: kontakty, pożyczone pieniądze, zapożyczone pieniądze i wirtualny portfel. Każdej encji odpowiada klasa opisująca jej atrybuty oraz relacje z innymi encjami. Dla porozumiewania się aplikacji z bazą danych stworzono interfejs użytkownika, który udostępnia podstawowe operacje dostępowe oraz wymagane funkcjonalności. Zestaw encji zapewnia podstawowe dane kontaktowe jak numer telefonu, dane osobowe oraz e-mail. Dodatkowo encja opisująca elektroniczny portfel umożliwia automatyczne dodawanie i odejmowanie zobowiązań z konta.

2.2 Ograniczenia

Z założeń wstępnych oraz ograniczenia złożoności projektu zawężono wysyłanie powiadomień mailowych za sprawą użytkownika. Wielkość danych przechowywanych w bazie oraz udostępnianych użytkownikowi nie jest ograniczona programowo. Aplikacja jako służąca do obsługi małego użytkownika nie zapewnia kontroli poprawności danych, więc możliwe jest wpisanie błędnych danych. W takim przypadku są one zapisywane, jednak program przeciwdziała występowaniu błędów z tego wynikających. W aktualnej formie aplikacja zapewnia obsługę jednego konta z walutą w PLN. Nie jest możliwe uruchomienie innej bazy danych w prosty sposób, należy wtedy ponownie ustawić migracje podczas kompilacji programu. Domyslnym i jedynym systemem operacyjnym, na którym aplikacja zadziała jest Windows obsługujący technologię .NET Framework 4.7.2 lub nowszy. Z celów czysto pragmatycznych ograniczono możliwości skalowania okienek.

3 Możliwości rozwoju aplikacji

Podstawową możliwością jest dodanie połączenia z dodatkowym zewnętrznym API udostępniającym dane o aktualnej dacie. W ten sposób możnaby wyświetlić użytkownikowi aktualny czas oraz zautomatyzować proces obsługi pożyczek, na przykład automatyczne wysyłanie powiadomień. Dodatkowym rozwinięciem mogłoby być wykorzystanie innych walut. Ostatnią prostszą możliwością skalowania programu jest wykorzystanie klasy portfeli w celu dywersyfikacji finansów, przykładowo różne waluty. W ten sposób możnaby również zwiększyć ilość użytkowników, co mogłoby być już dużą inwestycją czasową.