



POLITECHNIKA RZESZOWSKA

IM. IGNACEGO ŁUKASIEWICZA

WYDZIAŁ MATEMATYKI I FIZYKI STOSOWANEJ

ADRIAN JAKUBOWSKI

APLIKACJA DO NAUKI FISZEK

PROJEKT Z PRZEDMIOTU

USŁUGI SIECIOWE W BIZNESIE

Kierunek studiów:

Inżynieria i Analiza Danych, III rok

Rzeszów 2024

Spis treści

1	Temat projektu	2
2	Cel i zakres pracy	2
2.1	Cel projektu	2
2.2	Metodologia	3
3	Szczegółowy opis projektu	4
3.1	Menu główne	4
3.2	Dodawanie zestawu fiszek	4
3.2.1	Tworzenie zestawu fiszek	4
3.2.2	Dodawanie pojęć do nowoutworzonego zestawu	5
3.3	Dodawanie pojęć do istniejącego zestawu	7
3.4	Usuwanie fiszek oraz zestawów fiszek	8
3.4.1	Usuwanie zestawu	8
3.4.2	Usuwanie poszczególnych pojęć z zestawu	9
3.5	Nauka fiszek	10
3.6	Weryfikacja wiedzy - rozwiązywanie quizów	11
4	Potencjalny rozwój projektu	11
5	Wnioski i podsumowanie	12

1 Temat projektu

Tematem projektu z przedmiotu "Usługi sieciowe w biznesie" było utworzenie nowego rozwiązania, bądź przetestowanie istniejącego już oprogramowania w zakresie szeroko pojętego biznesu. Z powodu szerokiego wachlarza możliwości, jakie oferował nam ten projekt, mogliśmy się zdecydować na pracę mając do wyboru utworzenie/przetestowanie narzędzia, którego zadaniem jest np. optymalizacja pracy, poprawa wydajności, redukcja kosztów czy poprawa wiedzy oraz doświadczenia pracownika.

Szukając odpowiedniego tematu projektu, zdecydowałem się na stworzenie czegoś z czego sam będę miał możliwość oraz będę chciał korzystać w życiu codziennym. Z tego względu, zainspirowany portalem Quizlet, postanowiłem utworzyć prostą aplikację okienkową do nauki "wirtualnych fiszek", czyli komputerowych odpowiedników małych karteczek, skutecznych przy nauce m.in. słów oraz zwrotów w językach obcych czy definicji i pojęć z innych dziedzin.

2 Cel i zakres pracy

2.1 Cel projektu

Celem projektu, zgodnie z ustalonym tematem, było utworzenie aplikacji do nauki fiszek, dzięki którym użytkownik może uczyć się pojęć oraz definicji, analizować swoje wyniki, a następnie sprawdzać swoją wiedzę.

Głównym zadaniem takiej aplikacji jest możliwość tworzenia własnych zestawów fiszek, a następnie ich nauka, poprzez interaktywne "odwracanie" fiszki. Powodem, dla którego użytkownik chce korzystać z takiego narzędzia jest poprawa swojej wiedzy, więc konieczne było również zaimplementowanie możliwości sprawdzenia swojej wiedzy, poprzez opcję zaznacza, które pojęcia są już znane dla użytkownika, a które należy powtórzyć. Ponadto, zdecydowałem się na utworzenie także dodatkowej opcji w mojej aplikacji, a mianowicie quizu, który losuje pytanie spośród fiszek w wybranym zestawie oraz losuje także 4 odpowiedzi z czego jedna jest prawidłowa. W taki sposób, gracz może szlifować swoją wiedzę.

2.2 Metodologia

Spośród gąszcza narzędzi oraz języków programowania, dzięki którym mógłbym utworzyć moją aplikację, zdecydowałem się na język Python, a dokładnie na bibliotekę Tkinter, która pozwala na tworzenie prostych interfejsów graficznych. Umożliwia tworzenie okien, przycisków, etykiet i innych elementów interfejsu. Działanie w powyższej bibliotece opiera się głównie na dodawaniu i usuwaniu odpowiednich elementów okienka, przez co użytkownik ma wrażenie jakby korzystał z różnych podstron na stronie internetowej. Jeśli chodzi o tworzenie zestawów fiszek oraz ich przechowywanie, do tego celu zostały wykorzystane proste, niewielkiego rozmiaru słownikowe bazy danych JSON.



Zdecydowałem się na pracę w bibliotece Tkinter ze względu na prostotę tego narzędzia, dobre przystosowanie do tego konkretnego zadania oraz na wcześniejszą styczność z tym rozwiązaniem oraz dobre wspomnienia z nim związane.

3 Szczegółowy opis projektu

3.1 Menu główne

W menu głównym, użytkownik ma do wyboru kilka przycisków odnoszących się do bardziej rozwiniętych elementów aplikacji. Ponadto, możemy zauważyć także nazwę całej aplikacji - FISZING.



Z kwestii czysto technicznych, strona główna jest pierwszym oknem po inicjalizacji aplikacji. Program tworzy instancję klasy **FlashcardsApp** i przekazuje do niej obiekt **root**, jako główny widget biblioteki Tkinter. W funkcji inicjalizacyjnej znajdują się także fragmenty kodu poświęcone konfiguracji okna, związane m.in. z tytułem aplikacji, kolorami czy rozmiarem okna. Ponadto, tworzone są pierwsze przyciski, widoczne na stronie głównej, dzięki którym możemy przejść do innych "stron", a technicznie patrząc, do innych funkcji w kodzie.

3.2 Dodawanie zestawu fiszek

3.2.1 Tworzenie zestawu fiszek

Po kliknięciu przycisku **Utwórz nowy zestaw** użytkownik przechodzi do okna, w którym ma możliwość wpisania nazwy nowego zestawu. Widoczne są również przyciski do zatwierdzenia nowego zestawu oraz powrotu do strony głównej. Aby dodać estetyki swojej aplikacji, utworzyłem także placeholder w polu wprowadzania nazwy nowego zestawu.



Skupiając się na kodzie, po wybraniu przez użytkownika opcji **Utwórz nowy zestaw** w głównym oknie, uruchamiana jest metoda **make new set**, która tworzy pole do wpisania nazwy zestawu, a także zarządza placeholderem. Tak jak było wspomniane we wstępie, biblioteka Tkinter opiera się na usuwaniu i wyświetlaniu odpowiednich elementów. W tej metodzie usuwane są wszystkie przyciski ze strony głównej, aby kliknięcie powodowało przejście do pozornie "nowej strony".

3.2.2 Dodawanie pojęć do nowoutworzonego zestawu

Po zatwierdzeniu nazwy użytkownik otrzymuje możliwość do wpisywania pojęć oraz definicji do nowoutworzonego zestawu. Po potwierdzeniu hasła, powstaje okienko, które wyświetla obecnie zawarte pojęcia i ich definicje w danym zestawie.

The screenshot shows the FISZING application interface. At the top, the word "FISZING" is displayed in a large, stylized, black font. Below it, there are two input fields: "Wprowadz pojecie..." and "Wprowadz definicje...". Below these fields are two buttons: a green "Zatwierdz" button and a dark green "Powrót" button.

The screenshot shows the FISZING application interface after adding a flashcard. The "Wprowadz pojecie..." field contains the text "carrot" and the "Wprowadz definicje..." field contains the text "marchewka". The "Zatwierdz" button is highlighted in green. Below the buttons, a new list of flashcards is displayed in a light green box with a scroll bar. The list contains the following items: "dog: pies", "cat: kot", "rabbit: królik", and "hamster: chomik".

Potwierdzenie nazwy zestawu uruchamia funkcję **create new set**, dzięki której tworzony jest nowy plik JSON przechowujący pojęcia i definicje, a także wyświetlane są nowe pole do wpisywania haseł.

Po zatwierdzeniu hasła kod przechodzi do metody **add flashcard**, która pobiera (get) wpisane pojęcia, zapisuje je do zestawu (zarówno do pliku JSON jak i do okienka, które wyświetla się po zatwierdzenia hasła), a po wszystkim czysci pola inputu hasła. Metoda ta dodatkowo uruchamia metody **save flashcards** oraz **set in frame** odpowiedzialne odpowiednio za dopisywania hasła do pliku JSON oraz tworzenie ramki z pojęciami i wypisywanie do niej tych pojęć.

3.3 Dodawanie pojęć do istniejącego zestawu

Aby dodać pojęcia do zestawu, który został stworzony wcześniej, użytkownik korzysta z przycisku **Dodaj fiszki do zestawu** na stronie głównej. Następnie wyświetla się okno wyboru zestawu, a po podjęciu decyzji, w którym zestawie mają być dodane nowe hasła, wyświetla się okno analogiczne do tego, które pojawiała się po utworzeniu nowego zestawu.



Patrząc pod względem kodu, najpierw program przechodzi do funkcji **see all sets**, która jak sama nazwa wskazuje wyświetla użytkownikowi wszystkie zestawy (przy jednoczesnym usuwaniu odpowiednich przycisków). Za pomocą utworzonych w tej funkcji przycisków użytkownik ma możliwość przejścia do kolejnej funkcji - **show set flash-**

cards, która jako argument wejściowy przyjmuje nazwę zestawu i pozwala na dodawanie nowych pojęć do danego setu oraz wyświetla wszystkie pojęcia w ramce.

3.4 Usuwanie fiszek oraz zestawów fiszek

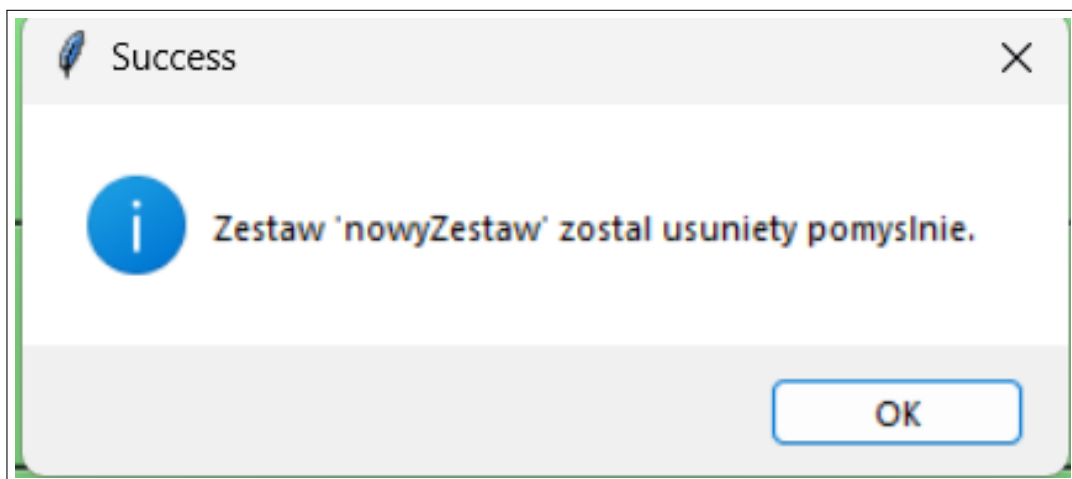
W takiej aplikacji konieczne jest również udostępnienie możliwości do usuwania zbędnych fiszek z określonych zestawów, jak i pozbywanie się całych setów haseł. Za pomocą przycisku **Usun..** użytkownik przechodzi do wyboru czy ma zamiar usunąć konkretną fiszkę z danego zestawu czy cały zestaw.



Program przechodząc do przycisku **Usun...** uruchamia funkcję **delete screen**, która ukrywa przyciski ze strony głównej oraz tworzy przyciski do wyżej opisanego wyboru.

3.4.1 Usuwanie zestawu

Jedną z opcji jest usuwanie całego zestawu fiszek. Użytkownik otrzymuje listę przycisków, dla każdego zestawu. Dodatkowo, utworzone zostały informacyjne message boxy, które zapobiegają przypadkowym usunięciom zestawów.



Po wyborze przycisku **Usun...** uruchamiana jest funkcja **delete screen**. Po tym, użytkownik wybierając opcję **Usun zestaw**, która przechodzi do funkcji **show delete sets**. Dzięki temu, osoba korzystająca z aplikacji ma możliwość wyboru zestawu do usunięcia, a gdy dokona wyboru, wyświetlany jest messagebox z pytaniem upewniającym, że jest to czyn zamierzony. Ponadto usuwany jest plik JSON dla danego pliku. Dzieje się to kolejno za pomocą funkcji **confirm delete set** oraz **delete set**.

3.4.2 Usuwanie poszczególnych pojęć z zestawu

Poza usunięciem całego zestawu, użytkownik ma możliwość usunięcia poszczególnych fiszek z wybranego zestawu. Podobnie jak w poprzednim przypadku, użytkownik najpierw wybiera zestaw, z którego chce usunąć fiszki, a następnie wyświetlana jest ramka z przyciskami odpowiadającymi konkretnemu pojęciu. Po kliknięciu wybranego hasła, usuwane jest ono z bazy, a ramka z przyciskami jest odświeżana. W tym przypadku,

pominięto funkcję messageboxów, gdyż potencjalna przypadkowa utrata pojedynczego hasła w bazie nie jest tak drastyczna. Ponadto, brak okienek dialogowych potwierdzających działanie, przyspiesza proces usuwania większej ilości fiszek.



Pod względem kodu, początek działania jest analogiczny do czynności usuwania całego zestawu. Zadanie polegające na usuwaniu poszczególnych pojęć zestawu rozpoczyna się w funkcji **delete cards**, gdzie użytkownik wybiera zestaw, z którego chce usunąć fiszki. Następnie w metodzie **show set flashcards delete** wyświetlana jest nazwa zestawu oraz następuje odwołanie do funkcji **delete flashcard in frame**, gdzie tworzona jest ramka wraz z przyciskami, które odpowiadają konkretnym pojęciom. Ostatecznie, usuwane jest pojęcie i definicja z bazy danych za sprawą metody **delete flashcard**.

3.5 Nauka fiszek

Najważniejszym zadaniem całej aplikacji jest nauka na podstawie "wirtualnych fiszek". Użytkownik dzięki aplikacji ma możliwość uczyć się poprzez swoiste "odwracanie" fiszek, a następnie zaznaczanie czy dane pojęcie zostało przez niego przyswojone czy jednak wymaga powtórki. Osoba ucząca się może w ten sposób dowiedzieć się ile haseł należy powtórzyć, aby osiągnąć stuprocentową wiedzę z danego zakresu.

Podobnie, jak w większości poprzednich funkcjonalności aplikacji, korzystający najpierw ma do wyboru zestaw, z którego chce się uczyć. Po wyborze, wyświetla się główne okno służące do nauki. Zawiera ono nazwę zestawu, licznik zagadnień oraz liczbę po-

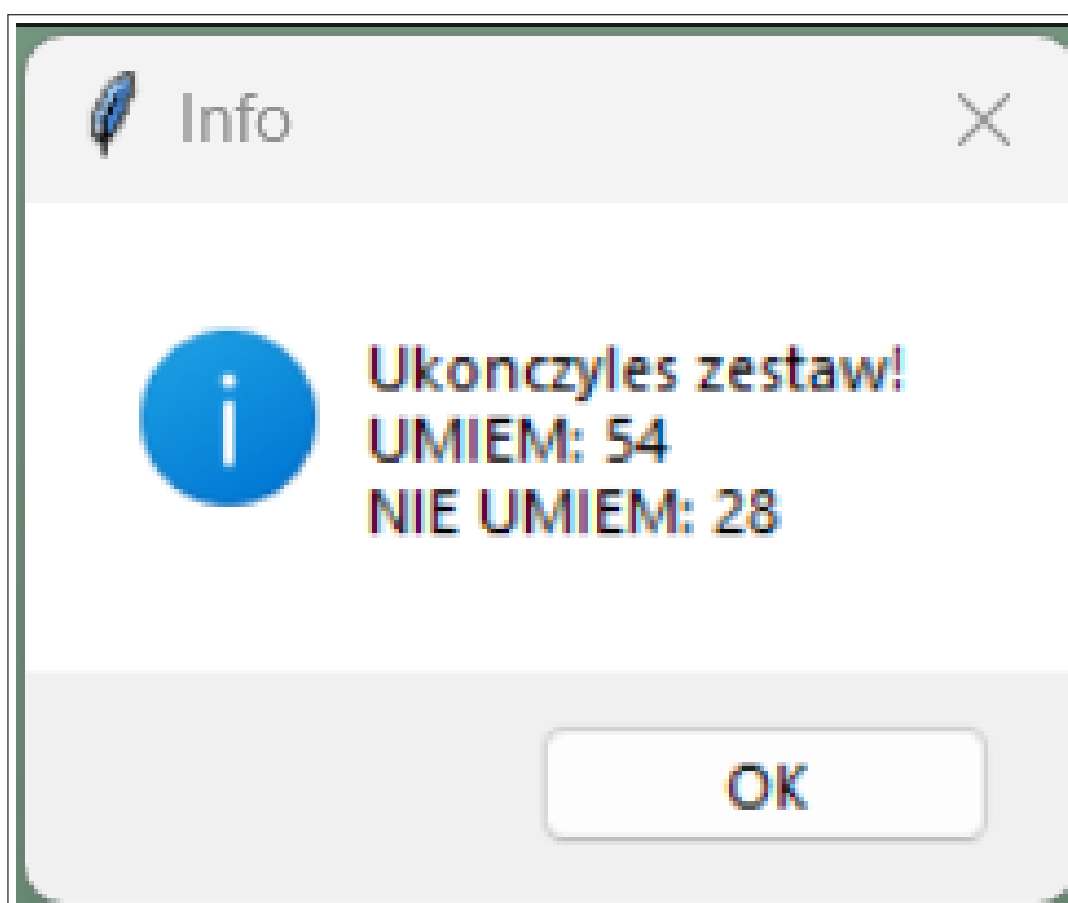
jęć, które użytkownik potrafi zapamiętać oraz ilość pojęć do powtórzenia. Ponadto, na samym środku jest duży przycisk, który służy do "odwracania" pojęcia, a także dwa dodatkowe przyciski, dzięki którym korzystający określa znajomość danego hasła.



Użytkownik przechodzi do kolejnego pojęcia poprzez wybór jednego z przycisków umiem/nie umiem. Iterowany jest także numer pojęcia oraz zwiększana jest liczba jednego z liczników służących do wyświetlania hasła przyswojonych oraz do powtórzenia.



Po dojściu do końca zestawu, wyświetla się krótkie podsumowanie ukazujące liczbę pojęć, które użytkownik potrafi oraz tych, które wymagają dalszej nauki.



3.6 Weryfikacja wiedzy - rozwiązywanie quizów

gfgfgf

4 Potencjalny rozwój projektu

gfhgfgf

5 Wnioski i podsumowanie

fdfdfar