



PROJEKT SŁOWNIKA Z POJĘCIAMI ARCHITEKTONICZNO - BUDOWLANymi

Projekt zaliczeniowy na kierunku Programista Python Developer

Rok: 2023 - 2024

Autor: Magdalena Koszela

Nr albumu: 91308

SPIS TREŚCI:

1. Opis projektu
2. Struktura kodu
 - a) Słownik
 - b) Program
3. Podsumowanie

1. OPIS PROJEKTU

Opisany w pracy program jest odpowiedzią na problem zaobserwowany w środowisku branży budowlanej związany z niepełnymi lub brakującymi definicjami stosowanymi na co dzień w procesie projektowym.

Wiele pojęć, które wymagają precyzyjnego ujęcia nie jest zawarta w ogólnodostępnych źródłach lub wymaga intensywnego poszukiwania w internecie i specjalistycznych źródłach wiedzy co powoduje trudności w pracy projektowej i wydłuża czas wykonywania obowiązków.

Słownik architektoniczno - budowlany miałby na celu usprawnić wyszukiwanie pojęć i posługiwanie się słownictwem zawartym w przepisach architektom, konstruktorom i innym inżynierom z branży architektoniczno - budowlanej.

Nie tylko dałby dostęp do szerokiej gamy pojęć ale również umożliwiałby dodawanie przez specjalistów kolejnych i edycję błędnych, ale jedynie po potwierdzeniu swojej tożsamości. Taki słownik miałby również cel dobroczynny. Za niewielką opłatą (która kierowana byłaby dla studentów branży budowlanej w krajach rozwijających się) pozwalałby wspomóc jakość edukacji najbardziej potrzebującym.

2. STRUKTURA KODU

a) Słownik

W pliku programu o nazwie „dictionary_WT.py” znajduje się podstawowy słownik z pojęciami architektonicznymi o nazwie „dictionary_WT” zawierający 26 kluczy i wartości.

Zawarte w nim dane bazują na ustawie:

OBWIESZCZENIE MINISTRA ROZWOJU I TECHNOLOGII 1)z dnia 15 kwietnia 2022 r.w sprawie ogłoszenia jednolitego tekstu rozporządzenia Ministra Infrastruktury w sprawie warunków technicznych, jakim powinny odpowiadać budynki i ich usytuowanie

(<https://isap.sejm.gov.pl/isap.nsf/DocDetails.xsp?id=WDU20220001225>)

b) Program

Następnie zdefiniowane zostają zastosowane funkcje, które przypisane są do poszczególnych czynności w aplikacji i znajdują się w poszczególnych plikach:

Plik: search_function.py

Kod rozpoczyna się od importu słownika z pliku dictionary_WT.py a następnie przechodzi do struktury funkcji:

Nazwa zawartej funkcji: find_keyword

Przeznaczenie: Funkcja do wyszukiwania pojęć w słowniku;

Parametry funkcji: „keyword”;

Działanie: Za pomocą warunku warunku „if” i operatora „in” sprawdza czy klucz istnieje w słowniku. Jeśli klucz istnieje w słowniku funkcja zwraca przypisaną mu wartość, jeśli nie istnieje – funkcja zwraca „None”.

Plik: add_function.py

Kod rozpoczyna się od importu słownika z pliku dictionary_WT.py a następnie przechodzi do struktury funkcji:

Nazwa zawartej funkcji: add_keyword

Przeznaczenie: funkcja służąca do dodawania pojęć do słownika.

Parametry funkcji: „keyword”, „definition”;

Działanie: Za pomocą warunku „if” i operatora „not in” sprawdza, czy dany klucz nie istnieje w słowniku i dodaje nowy klucz z przypisaną wartością do słownika. Zwraca informację o dodaniu pojęcia. Jeśli klucz istnieje już w słowniku funkcja zwraca informację o istniejącym kluczu.

Plik: edit_function.py

Kod rozpoczyna się od importu słownika z pliku dictionary_WT.py a następnie przechodzi do struktury funkcji:

Nazwa zawartej funkcji: edit_keyword

Przeznaczenie: Funkcja służąca do edycji istniejących pojęć w słowniku.

Parametry funkcji: „keyword”, „new_definition”;

Działanie: Za pomocą warunku „if” i operatora „in” sprawdza czy klucz jest w słowniku. Jeśli klucz znajduje się w słowniku, do istniejącego klucza zostaje przypisana nowa definicja. Funkcja zwraca informację, że definicja została zaktualizowana. Jeśli słownik nie zawiera wskazanego klucza funkcja zwraca informację, że w słowniku nie ma takiej definicji.

Plik: delete_function.py

Kod rozpoczyna się od importu słownika z pliku dictionary_WT.py a następnie przechodzi do struktury funkcji:

Nazwa zawartej funkcji: delete_keyword

Przeznaczenie: Funkcja służąca do zgłoszenia pojęć do usunięcia pojęć w przypadku ich zdezaktualizowania.

Parametry funkcji: „keyword”;

Działanie: Za pomocą warunku „if” i operatora „in” sprawdza czy klucz jest w słowniku. Jeśli klucz znajduje się w słowniku, funkcja zwraca informację, że pojęcie zostało zgłoszone do usunięcia i usuwa klucz i wartość ze słownika. Jeśli klucz nie istnieje w słowniku funkcja zwraca informację, że klucz nie istnieje w słowniku.

Plik: login_function.py

Nazwa zawartej funkcji: login_panel

Przeznaczenie: Funkcja pozyskująca dane do rejestracji nowego użytkownika.

Elementy funkcji: zmienne: name, surname, license_number, email, login, password, card_number, payment;

Działanie: Za pomocą funkcji „input” pobiera poszczególne informacje od użytkownika.

Dla każdej zmiennej utworzona jest pętla while, która powoduje zapętlenie wprowadzania danych przez użytkownika do czasu aż nie wprowadzi poprawnej wartości.

Dla zmiennej *name* i *surname* warunkiem poprawności jest długość stringu większa od 0.

Dla zmiennej *license_number* warunkiem poprawności jest długość stringu większa od 0 i string zawierający tylko znaki alfanumeryczne.

Zmienna *email* musi zawierać znak „@” i kończyć się wyrażeniem „.com” lub „.pl”.

Zmienna *login* i *password* przyjmuje jedynie stringi o długości kolejno max. 8 i 6 znaków i długość stringu większą od 0.

Zmienna *card_number* przyjmuje jedynie wartości liczbowe składające się z 15 cyfr.

Zmienna *payment* przyjmuje jedynie wartości integer.

W przypadku wpisania wartości w formie stringu wyjątek definiuje rozwiązanie „ValueError” przez informowanie użytkownika o konieczności wpisania wartości liczbowej. Jeśli następuje wpisanie wartości mniejszej lub równej 0 użytkownik otrzymuje informację o wymaganej wartości dodatniej.

Po wprowadzeniu wszystkich danych poprawnie zwraca informację o utworzeniu konta.

Plik: user_login_function.py

Nazwa zawartej funkcji: user_login

Przeznaczenie: Funkcja pozyskująca dane do logowania użytkownika.

Elementy funkcji: zmienne: license_number, login, password;

Działanie: Za pomocą funkcji „input” pobiera informacje od użytkownika. Dla zmiennej *license_number* warunkiem poprawności jest długość stringu większa od 0 i string zawierający tylko znaki alfanumeryczne.

Zmienna *login* i *password* przyjmuje jedynie stringi o długości kolejno max. 8 i 6 znaków i długość stringu większą od 0. Po wprowadzeniu wszystkich danych poprawnie zwraca informację o zalogowaniu.

Plik: next_choice_function.py

Nazwa zawartej funkcji: next_choice

Przeznaczenie: Funkcja umożliwiająca wybór kolejnego działania na słowniku.

Działanie: Za pomocą funkcji „input” pobiera od użytkownika numer, który przypisany jest do konkretnego działania. Jeśli wybór to 1, 2, 3, 4, 5, 6 lub 7 funkcja zwraca podany numer. Jeśli nie, zwraca informację o nieprawidłowym wyborze do czasu aż wybrany numer nie będzie mieścić się w przedziale.

Plik: menu_function.py

Kod rozpoczyna się od importu słownika z pliku dictionary_WT.py oraz importu wszystkich wymienionych powyżej funkcji.

Nazwa zawartej funkcji: interactive_dictionary_WT

Przeznaczenie: Funkcja łącząca wszystkie pozostałe funkcje i tworząca słownik umożliwiający wyszukiwanie, modyfikowanie i wprowadzanie danych.

Działanie: Na początku następuje zdefiniowanie zmiennej logged_in jako False, by do opcji słownika takich jak wprowadzanie nowych pojęć i edycja istniejących definicji mieli użytkownicy, którzy podadzą swój login i hasło.

Podczas uruchomienia słownik wyświetla hasło powitalne, opis i menu następnie za pomocą funkcji „input” pobiera od użytkownika wskazaną opcję:

1) choice == 1: 1. Wyszukaj pojęcie:

Jeżeli wybrana opcja to 1 to funkcja input umożliwia wprowadzenie szukanego pojęcia w słowniku.

Argument za pomocą funkcji **find_keyword** znajduje szukaną definicję jeśli klucz istnieje w słowniku.

Następuje uruchomienie funkcji **next_choice**.

Jeśli klucza nie ma w słowniku zwraca informację, że słowo nie zostało znalezione w słowniku. Funkcja umożliwia ponowne wpisanie szukanego słowa aż do jego znalezienia.

2) choice == 2: 2. Dodaj nowe pojęcie:

Jeśli przed wyborem opcji 2 wybraliśmy opcję 6 to funkcja umożliwia nam dodanie do słownika nowego klucza i wartości za pomocą funkcji **add_keyword**. Następuje uruchomienie funkcji **next_choice**.

Jeżeli logged_in = False to funkcja wyświetla komunikat o wymogu zalogowania się.

3) choice == 3: 3. Edytuj istniejące pojęcie:

Jeśli przed wyborem opcji 3 wybraliśmy opcję 6 to funkcja umożliwia nam przypisanie nowej wartości do istniejącego klucza za pomocą funkcji **edit_keyword**. Następuje uruchomienie funkcji **next_choice**.

Jeśli klucz nie istnieje w słowniku, funkcja zwraca informację o jego braku.

Jeżeli logged_in = False to funkcja wyświetla komunikat o wymogu zalogowania się.

4) choice == 4: 4. Zgłoś pojęcie do usunięcia:

Jeśli przed wyborem opcji 4 wybraliśmy opcję 6 to funkcja zwraca nam informację o zgłoszeniu definicji do usunięcia za pomocą uruchomionej funkcji **delete_keyword**. Następuje uruchomienie funkcji **next_choice**.

Jeśli klucz nie istnieje w słowniku, funkcja zwraca informację o jego braku.

Jeżeli `logged_in = False` to funkcja wyświetla komunikat o wymogu zalogowania się.

5) `choice == 5`: 5. Zarejestruj się:

Następuje uruchomienie funkcji **login_panel** i pobranie od użytkownika informacji takich jak imię, nazwisko, numer uprawnień, email, login, hasło, numer karty i wysokość wpłaty.

Po wprowadzeniu danych zwracany jest komunikat o utworzeniu konta.

Następuje uruchomienie funkcji **next_choice**.

6) `choice == 6`: 6. Zaloguj się:

Za pomocą funkcji **user_login** pobierane są dane logowania od użytkownika. Po wpisaniu danych funkcja zwraca komunikat o zalogowaniu się. Zmienna `logged_in` otrzymuje wartość `True` co pozwala na działania w opcjach 2, 3 i 4.

6) `choice == 7`: 7. Wyjdź:

Opcja kończy działanie programu.

Po każdym wykonanym poleceniu program oczekuje na „enter”. Następnie przechodzi do funkcji „next_choice”, która umożliwia wybór kolejnego polecenia(polecenie należy wpisać dwukrotnie).

Plik główny uruchamiający program:

Słownik_architektoniczno-budowlany

Plik składa się z polecenia importu pliku `menu_function` i uruchomienia zawartej w nim funkcji `interactive_dictionary_WT`.

3. PODSUMOWANIE

Przedstawiony program umożliwia podstawowe działania na słowniku takie jak wyszukiwanie, wprowadzanie, edytowanie czy usuwanie pojęć. Baza słownika składa się jedynie z kilku haseł aczkolwiek zadaniem użytkowników jest rozwinięcie go i dostosowanie do swoich potrzeb. Wymóg potwierdzenia tożsamości przez wpisanie loginu, hasła i nr uprawnień zmniejsza możliwość wprowadzenia pojęć przez osoby niepowołane i nie mające wystarczającej wiedzy zawodowej.

Biorąc pod uwagę powyższe, słownik będzie ułatwieniem dla projektantów w poszukiwaniu rzadziej spotykanych i trudno dostępnych pojęć i usprawni pracę projektową wszystkim branżom.