Dokumentacja Techniczna Aplikacji do Zarządzania Rezerwacjami

Autor : Bartosz Jan Pękała

Data : 27.10.2024 r.

Spis treści

[1. Wstęp 3](#_Toc181035028)

[2. Opis Projektu 4](#_Toc181035029)

[3. Wymagania techniczne 5](#_Toc181035030)

[4. Instrukcja uruchomienia 5](#_Toc181035031)

[5. Struktura techniczna projektu 7](#_Toc181035032)

[6. Podsumowanie 14](#_Toc181035033)

# Wstęp

Niniejsza dokumentacja techniczna opisuje aplikację do zarządzania rezerwacjami, która została stworzona przy użyciu języka programowania Python oraz bazy danych PostgreSQL. Aplikacja została zaprojektowana w celu umożliwienia użytkownikom łatwego zarządzania terminami spotkań i rezerwacjami. Główne funkcje aplikacji obejmują dodawanie nowych rezerwacji, sprawdzanie dostępności terminów oraz przeglądanie istniejących spotkań.

Aplikacja opiera się na graficznym interfejsie użytkownika stworzonym z użyciem biblioteki Tkinter, co zapewnia intuicyjną obsługę oraz prostą nawigację. PostgreSQL pełni rolę głównej bazy danych, przechowującej informacje o wszystkich rezerwacjach, co gwarantuje skalowalność i bezpieczne przechowywanie danych.

Dokumentacja zawiera instrukcje dotyczące instalacji i uruchomienia projektu i opis jego struktury

# Opis Projektu

Aplikacja do zarządzania rezerwacjami jest narzędziem służącym do obsługi i organizacji spotkań oraz rezerwacji terminów. Została zaprojektowana tak, aby umożliwić użytkownikom szybkie dodawanie nowych rezerwacji, sprawdzanie dostępności terminów oraz przeglądanie istniejących spotkań. Aplikacja posiada intuicyjny interfejs graficzny, który umożliwia łatwą nawigację.

Główne cechy aplikacji:

* Dodawanie nowych rezerwacji z losowym numerem spotkania, z jednoczesnym sprawdzeniem dostępności wybranego przedziału czasowego.
* Sprawdzanie dostępności terminów o wybranej dacie oraz przedziale czasowym.
* Przeglądanie istniejących spotkań w formie tabeli.

Aplikacja została stworzona przy użyciu języka programowania Python oraz biblioteki Tkinter do obsługi interfejsu graficznego. Za przechowywanie danych odpowiada baza danych PostgreSQL, która zapewnia skalowalność oraz bezpieczne przechowywanie informacji o rezerwacjach.

# Wymagania techniczne

* Python (wersja 3.8 lub wyższa)
* Biblioteki Pythona:
  + Tkinter (standardowa biblioteka w Pythonie)
  + Psycopg2 (psycopg2-binary)
* PostgreSQL (wersja 12 lub wyższa)
* pgAdmin (zarządzanie bazą danych graficznie)

# Instrukcja uruchomienia

Uruchomienie aplikacji przy użyciu konsoli

1. Upewnij się, że wszystkie wymagania techniczne są spełnione, w tym instalacja Pythona i bazy danych PostgreSQL.
2. Otwórz terminal lub wiersz poleceń.
3. Przejdź do katalogu, w którym znajduje się plik aplikacji (np. main.py).
4. Uruchom aplikację za pomocą polecenia: *python main.py*
5. Po uruchomieniu aplikacji zobaczysz okno interfejsu graficznego.

Uruchomienie aplikacji w Visual Studio Code

1. Upewnij się, że masz zainstalowany Visual Studio Code oraz Python na Twoim komputerze.
2. Otwórz Visual Studio Code.
3. Wybierz „Plik” -> „Otwórz folder” i wskaż folder, w którym znajduje się Twój plik skryptu (np. *main.py*).
4. Kliknij dwukrotnie na plik *main.py*, aby otworzyć go w edytorze.
5. Aby uruchomić skrypt, kliknij prawym przyciskiem myszy w obrębie edytora kodu i wybierz opcję „Run Python File in Terminal” lub naciśnij klawisz F5 albo kliknij przycisk „Run”.
6. Skrypt uruchomi się w zintegrowanym terminalu w Visual Studio Code. Zobaczysz okno interfejsu graficznego aplikacji.

Przeglądanie istniejących spotkań

1. Kliknij przycisk „Odczytaj dane z bazy”.
2. Aplikacja pobierze i wyświetli wszystkie istniejące rezerwacje w formie tabeli.
3. Możesz przeglądać tabelę i zobaczyć szczegóły każdego spotkania.

Dodawanie nowej rezerwacji

1. W oknie aplikacji wybierz odpowiednią datę z listy rozwijanej „Data”.
2. Wybierz dostępny przedział czasowy z listy rozwijanej „Przedział czasowy”.
3. Kliknij przycisk „Zapisz dane do bazy”.
4. Aplikacja sprawdzi, czy wybrany przedział czasowy jest wolny, i jeśli tak, zarezerwuje termin z unikalnym numerem spotkania.
5. Po dodaniu rezerwacji, tabela zostanie zaktualizowana, aby pokazać nowo dodane spotkanie.

Sprawdzanie dostępności terminu

1. Wybierz odpowiednią datę z listy rozwijanej „Data”.
2. Wybierz przedział czasowy z listy rozwijanej „Przedział czasowy”.
3. Kliknij przycisk „Sprawdź dostępność”.
4. Aplikacja sprawdzi, czy wybrany termin jest wolny. Jeśli jest zajęty, wyświetli istniejące spotkanie w tabeli.
5. Jeśli termin jest wolny, wyświetli odpowiedni komunikat.

Zamknięcie aplikacji

1. Aby zakończyć pracę z aplikacją, zamknij okno aplikacji, klikając przycisk zamknięcia okna (X) w prawym górnym rogu.
2. Jeśli uruchomiłeś aplikację w terminalu, zakończ pracę, naciskając Ctrl+C w terminalu.

# Struktura techniczna projektu

Architektura aplikacji

Aplikacja została zaprojektowana w oparciu o model klient-serwer, gdzie interfejs użytkownika (GUI) pełni funkcję klienta, a baza danych PostgreSQL pełni funkcję serwera danych. Komunikacja między aplikacją a bazą danych odbywa się za pomocą zapytań SQL, obsługiwanych przez bibliotekę psycopg2.

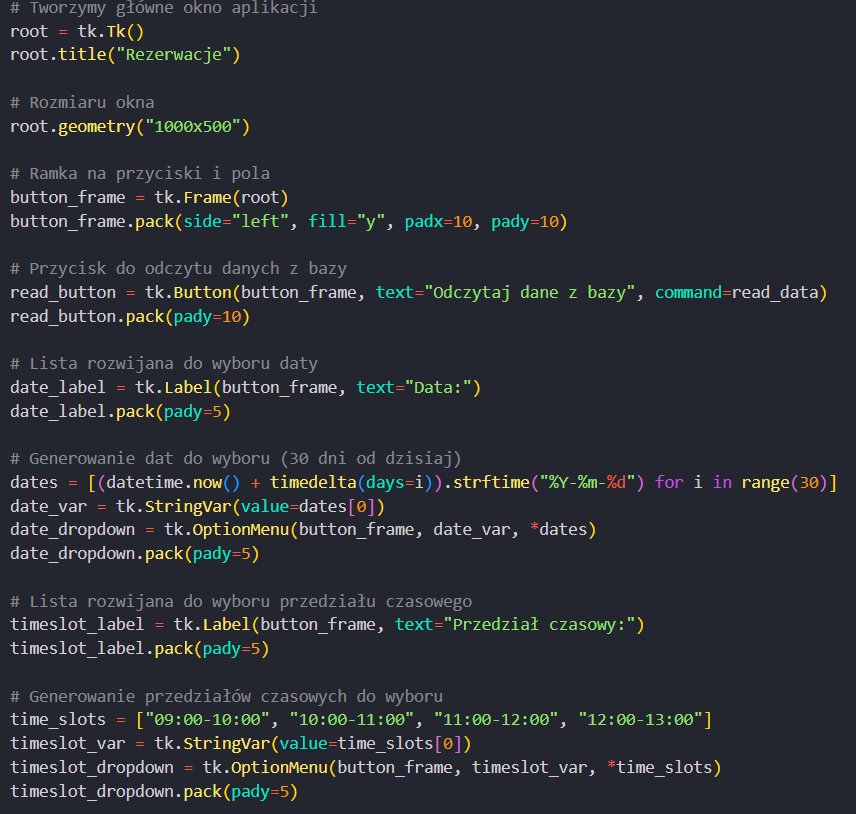
Technologie użyte w projekcie

* Python: Główny język programowania użyty do stworzenia aplikacji.
* Tkinter: Biblioteka GUI Pythona używana do tworzenia interfejsu graficznego aplikacji.
* PostgreSQL: Baza danych relacyjna, w której przechowywane są informacje o rezerwacjach.
* Psycopg2: Biblioteka Pythona służąca do połączenia z bazą danych PostgreSQL.

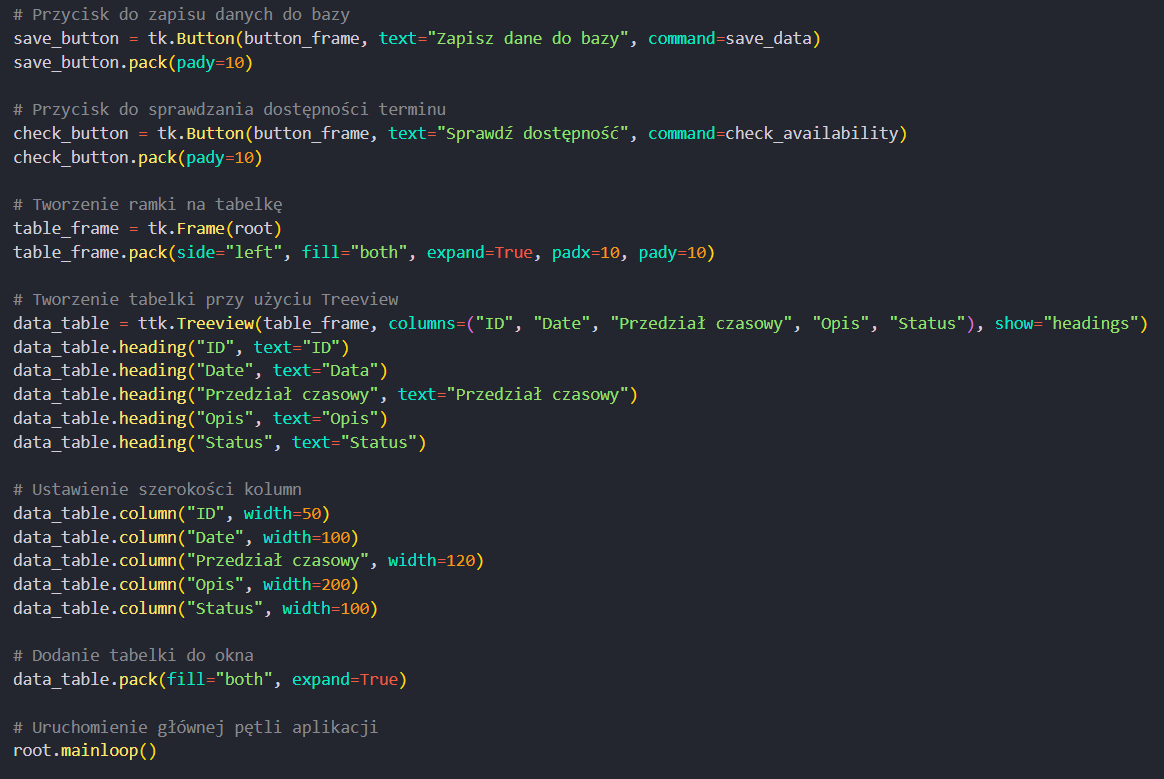
Struktura kodu

Kod źródłowy aplikacji podzielony jest na kilka modułów odpowiadających za różne funkcje:

1. Moduł GUI (Tkinter): Odpowiada za tworzenie interfejsu graficznego, w tym przycisków, pól wprowadzania, list rozwijanych oraz tabeli do wyświetlania danych.

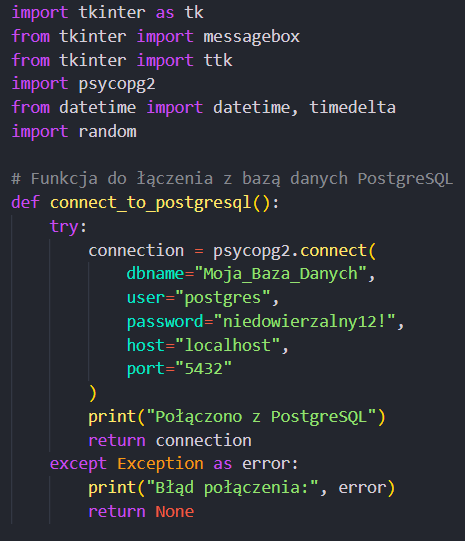


Rysunek 1 Kod tworzenia interfejsu cz. 1

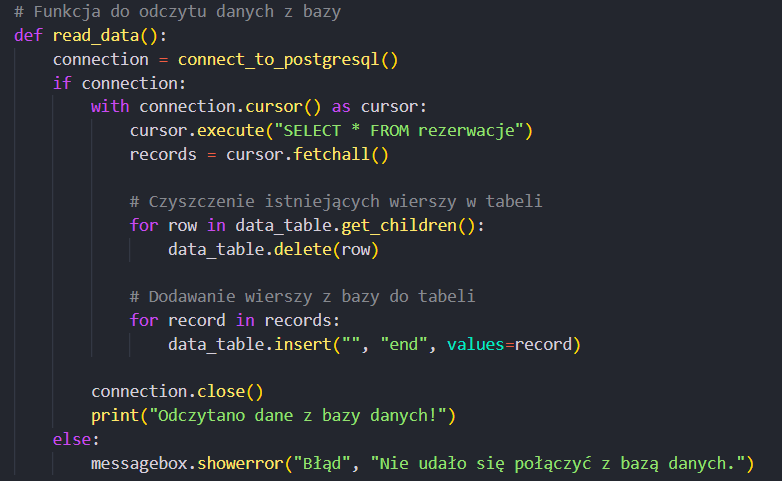


Rysunek 2 Kod tworzenia interfejsu cz. 2

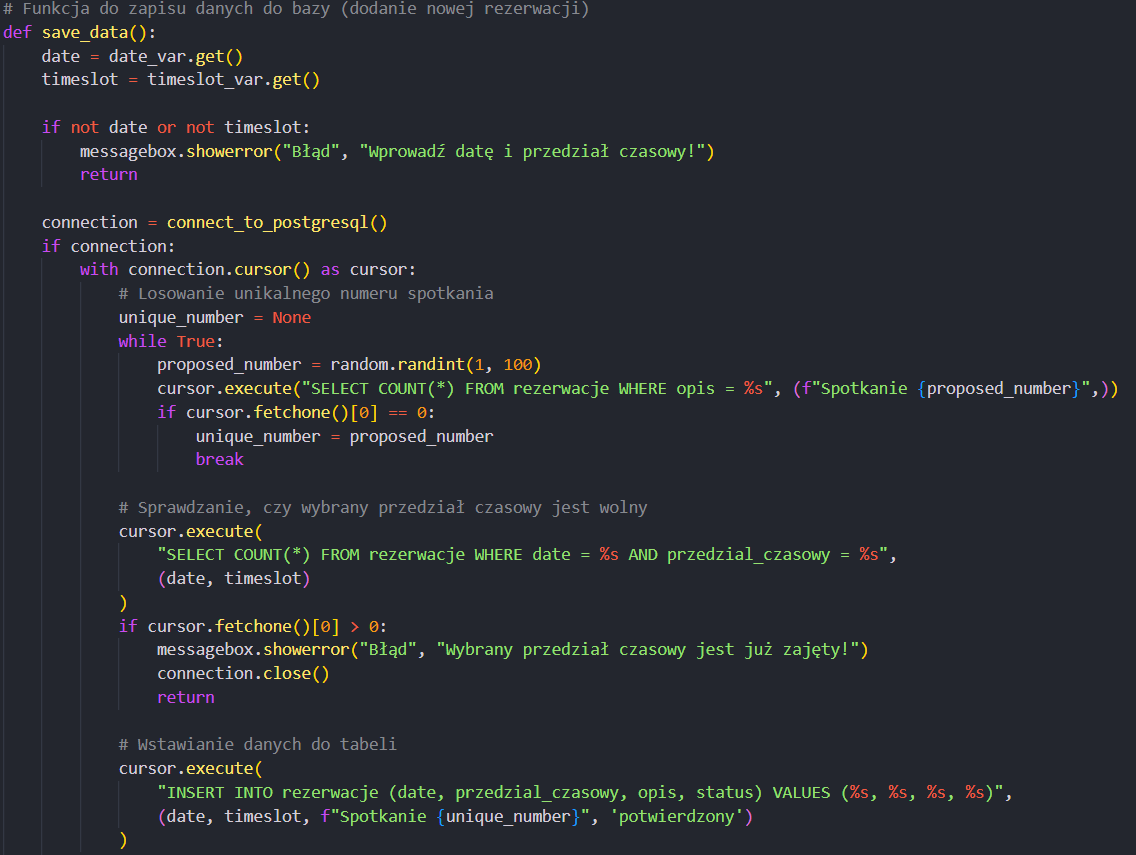
1. Moduł obsługi bazy danych (Psycopg2): Zawiera funkcje odpowiedzialne za połączenie z bazą danych, pobieranie i zapisywanie danych, oraz wykonywanie operacji SQL.

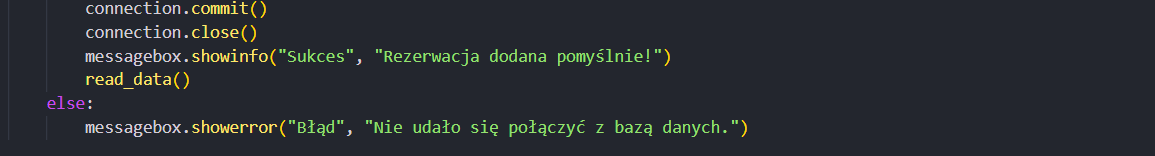


Rysunek 3 Biblioteki i łączenie z bazą danych



Rysunek 4 Funkcja odczytu bazy danych



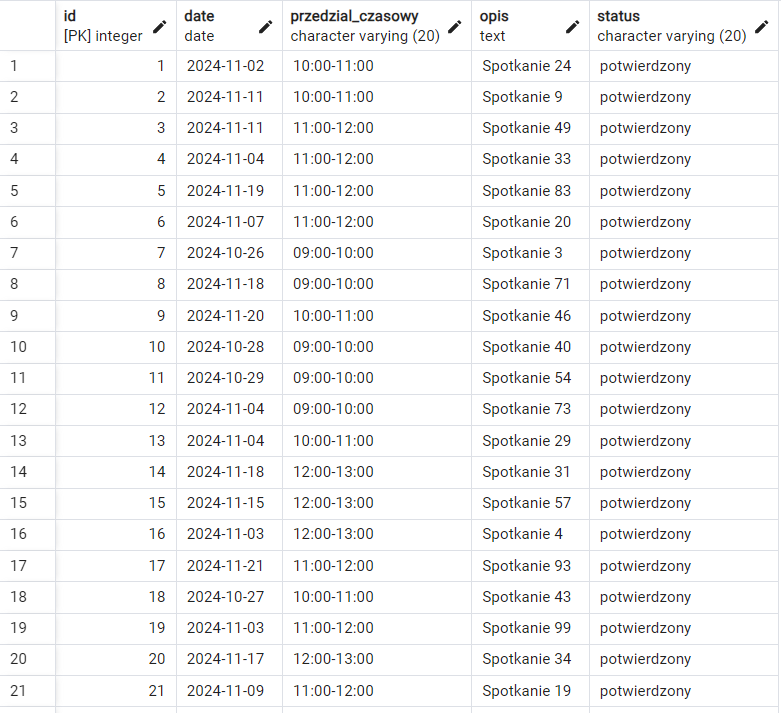


Rysunek 5 Funkcja zapisywania rezerwacji do bazy danych



Rysunek 6 Funkcja sprawdzania dostępności rezerwacji w danym dniu i czasie

1. Tabela rezerwacje w bazie danych: Przechowuje informacje o wszystkich rezerwacjach, takie jak data, przedział czasowy, numer spotkania i status.



Rysunek 7 Tabela rezerwacji

# Podsumowanie

Aplikacja do zarządzania rezerwacjami została stworzona jako intuicyjne narzędzie do organizowania spotkań i terminów. Dzięki wykorzystaniu języka Python, biblioteki Tkinter do tworzenia interfejsu graficznego oraz bazy danych PostgreSQL, projekt zapewnia skalowalne i elastyczne rozwiązanie do zarządzania danymi.

Główne funkcjonalności aplikacji obejmują dodawanie nowych rezerwacji, sprawdzanie dostępności terminów oraz przeglądanie istniejących spotkań w formie tabeli. Dzięki modułowej budowie i zastosowaniu nowoczesnych technologii, aplikacja jest łatwa w rozbudowie oraz modyfikacji w celu dodania nowych funkcji lub obsługi większej liczby użytkowników.

Projekt został stworzony z myślą o łatwości obsługi, intuicyjnej nawigacji oraz bezpiecznym zarządzaniu danymi. W przyszłości aplikacja może zostać rozwinięta o dodatkowe funkcje, takie jak powiadomienia o rezerwacjach czy integracja z zewnętrznymi kalendarzami.