

Wstęp do sprawozdania z projektu z przedmiotu PJF

Julia Ołowniuk WCY21KA1S1

Nr Albumu 80968

ToDo List/Planner

Aplikacja okienkowa ułatwiająca planowanie

Github: https://github.com/JuliaOłowniuk/PJF_Ołowniuk

Ogólny opis aplikacji:

ToDo List/Planner to aplikacja okienkowa napisana w języku Python (Flask) , stworzona w celu ułatwienia organizacji codziennych zadań i planowania czasu. Pozwala użytkownikowi tworzyć listę zadań, określać priorytety, ustawiać terminy, odznaczać zadania czy kopiować plany z innych dni.

Przykładowa użyteczność aplikacji:

- Organizacja Zadań: Efektywna organizacja zadań na różne okresy czasowe, możliwy wybór widoków tygodniowy/ miesięczny/ roczny
- Odznaczanie wykonanych już zadań- widzimy ile nam zostało, mogą jednak te które już wykonaliśmy pozostać zaciemnione np. aby wiedzieć ile już zrobiliśmy – psychologicznie działałoby to tak, że mamy poczucie, że wykonaliśmy naprawdę sporo i możemy być z siebie dumni
- Priorytetyzacja: Możliwość przypisywania priorytetów zadaniom dla skupienia na najważniejszych.
- - Kategorie: Można kolorami zaznaczać kategorie np. dom, szkoła, siłownia etc .i patrząc na dany dzień będzie można zapamiętywać łatwiej dzięki temu jakie zadania danego dnia mamy do wykonania. Przypisanie do kategorii będzie odbywało się podczas wprowadzania zadań
- Terminy: Dodawanie i monitorowanie terminów wykonania zadań.
- Powiadomienia: Ustawianie powiadomień o zbliżających się terminach.
- Auto-Pomocnik- wprowadzam jaką chcę wykonywać czynność ile ona czasu będzie trwała, a planner proponuje kiedy można w zadanym przedziale czasowym wykonać
- Statystyki/ wykresy/ raporty- kontrolowanie na podstawie kategorii, priorytetów i właśnie z tych działów jako miernik naszej efektywności i taki realny rezultat naszego dysponowania czasem, do czego mamy więcej predyspozycji, co pomoże z pewnością poprawić nasze nawyki i może więcej zajmować się rzeczami z ważniejszym priorytetem.
- Logowanie się/ założenie własnego konta w aplikacji – po to aby móc zalogować się na innym urządzeniu i wciąż mieć swój planner
- Autouzupełnianie z innych dni – przydatna opcja, jeśli plany się powtarzają

Funkcje aplikacji:

- Dodawanie Zadań: Dodawanie zadań z nazwą, opisem, priorytetem i terminem.
- Edytowanie/Zaznaczanie Zadań jako Zrobione: Modyfikacja danych zadań i oznaczanie jako wykonane.
- Usuwanie Zadań: Usuwanie niepotrzebnych zadań.
- Sortowanie Zadań: Sortowanie według priorytetu, terminu, itp.
- Wyszukiwanie: Szybkie odnajdywanie konkretnego zadania.
- Powiadomienia: Automatyczne powiadomienia o zbliżających się terminach.
- Statystyki: Wyświetlanie statystyk dotyczących ilości zadań, wykonanych zadań, itp.
- Logowanie/ Rejestracja: Możliwość posiadania swoich danych na kilku kontach
- Udostępnianie: udostępnianiu planu dnia/ tygodnia/ miesiąca/ roku

Przykładowe biblioteki, które prawdopodobnie zostaną użyte do projektu:

- Tkinter: Tworzenie interfejsu okienkowego użytkownika. / lub PyQt(5 lub 6) właśnie do przypisywania kategorii i priorytetów do zadań(dzięki temu łatwo sortować i dawać priorytety do zadań)
- SQLite (sqlite3): Przechowywanie danych o zadaniach (wbudowana baza danych)
- datetime: Obsługa danych dotyczących terminów.
- tkinter.messagebox: Wyświetlanie powiadomień/ komunikaty. / lub pymsgbox
- threading: Do obsługi wielowątkowości. Można użyć wątków do obsługi równoczesnych operacji, takich jak aktualizacja interfejsu i generowanie powiadomień bez blokowania interakcji użytkownika.
- matplotlib/ pandas: Do generowania wykresów i statystyk oraz raportów. Matplotlib / Pandas pozwala wizualizować dane, co jest przydatne do analizy czasu i postępu w wykonywaniu zadań.
- Flask-Login i Flask-WTF (do aplikacji opartych na Flask):
- Flask-Login: Umożliwia obsługę systemu logowania.
- Flask-WTF: Dostarcza narzędzia do obsługi formularzy, co jest przydatne przy implementacji formularzy logowania i rejestracji.
- Flask- do udostępniania z aplikacji
- schedule:- schedule to prosta biblioteka do planowania zadań w Pythonie.
/ lub APScheduler:
- APScheduler to biblioteka do zarządzania harmonogramem zadań w Pythonie. Pozwala na dodawanie, usuwanie i planowanie zadań w różnych formatach czasu.
- time- funkcji do zarządzania czasem i obsługi zadań związanych z czasem