A book with a globe inside

Description automatically generatedLearn-ey

Spis treści

[1. Opis celu projektu 2](#_Toc128662469)

[2. Wymagania funkcjonalne 3](#_Toc128662470)

[3. Wymagania niefunkcjonalne 3](#_Toc128662471)

[4. Opis wybranego środowiska programistycznego 4](#_Toc128662472)

[5. User stories 7](#_Toc128662473)

[6. Narzędzia pomocnicze 8](#_Toc128662474)

[7. Product backlog 9](#_Toc128662475)

1. **Opis celu projektu**

\*tu wstaw nazwę\*/Learn-ey to aplikacja desktopowa wspomagająca proces nauki. Celem projektu jest stworzenie narzędzia, które pozwoli użytkownikowi na skuteczniejsze przyswajanie wiedzy z wybranej przez siebie dziedziny lub słówek w obcych językach. Program będzie charakteryzował się prostotą użytkowania i minimalistycznym interfejsem. Podczas korzystania z aplikacji użytkownik będzie miał możliwość wprowadzić do aplikacji materiał, który chce przyswoić za pomocą formularza   
lub za pomocą wcześniej przygotowanego pliku, z którego aplikacja wczyta materiały. Nauka odbywać się będzie w oparciu o metodę pomodoro - sposób zarządzania czasem ukierunkowany na zwiększenie produktywności oraz koncentracji, polega na rozbiciu procesu nauki na sekcje: 25 min intensywnego przyswajania wiedzy i odpoczynek przez kolejne 5 min. Następnie należy powtórzyć cały proces.   
Po 4 pełnych cyklach następuje dłuższa przerwa zazwyczaj 15 do 30 minut. Zadania, poprzez które następuje nauka to:

1. Gra w wisielca
2. Fiszki
3. Prawda/Fałsz
4. Wskaż poprawną definicję
5. Wskaż poprawne słowo
6. Dopasuj definicję do zagadniania

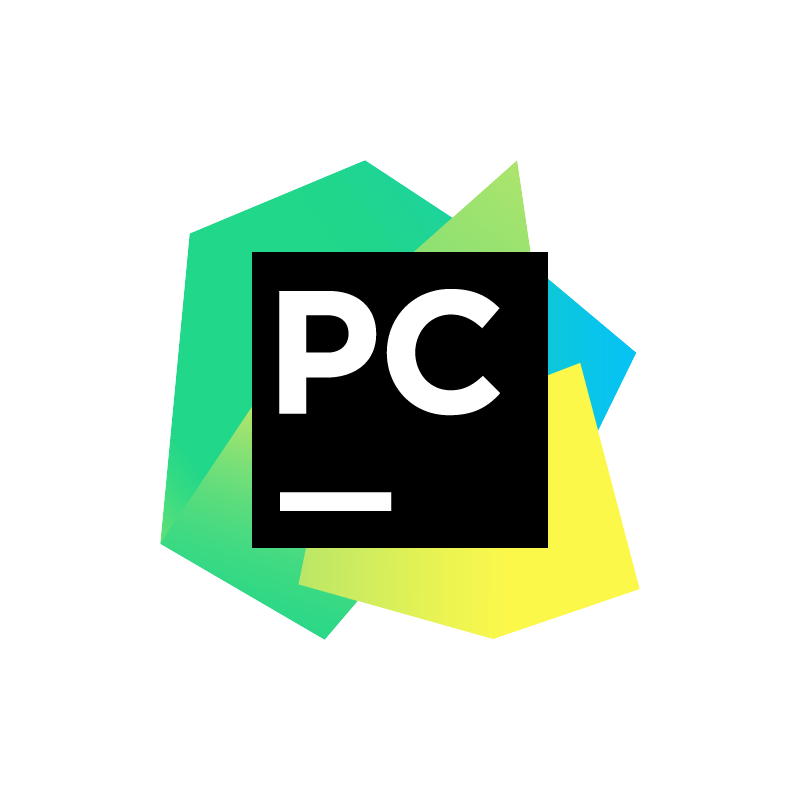
Użytkownik po każdej odbytej sesji nauki otrzyma raport z wygenerowany na podstawie uzyskanych przez program odpowiedzi.

Projekt napisany zostanie w języku Python ze względu na jego intuicyjność, przejrzystość oraz mnogość dostępnych modułów, które mogą znacznie ułatwić implementację części rozwiązań mających znaleźć zastosowanie w projekcie.

1. **Wymagania funkcjonalne**

**Aplikacja będzie spełniać następujące funkcje:**

1. Wprowadzanie materiałów do nauki:
   1. Wprowadzanie zagadnień do nauki za pomocą formularza.
   2. Wprowadzenie zagadnień do nauki za pomocą pliku.
2. Wybór języka do nauki popularnych słów.
   1. Wyświetlenie menu z dostępnymi językami:
   2. Generowanie 50 najpopularniejszych słówek w wybranym języku.
3. Nauka wybranego materiału:
   1. Podział materiału na sekcje.
   2. Przetworzenie materiału na zadanie interaktywne i fiszki.
   3. Wygenerowanie funkcji zliczającej punkty.
   4. Generowanie zadań interaktywnych w odpowiednym momencie nauki.
4. Generowanie raportów po zakończeniu sesji.
5. Dostosowanie czasu zegara pomodoro.
6. **Wymagania niefunkcjonalne**
7. Aplikacja powinna działać na komputerach korzystających z systemów Windows 10, Windows 11, Linux oraz MacOS.
8. Interfejs powinien wyświetlać się w trybie okienkowym.
9. Intuicyjna obsługa oraz czytelność.
10. Aplikacja do poprawnego działania powinna mieć dostęp do internetu.
11. **Opis wybranego środowiska programistycznego**



PyCharm to zintegrowane środowisko programistyczne stworzone przez JetBrains   
dla programistów Python. Jest to komercyjny produkt, ale istnieje również darmowa wersja, PyCharm Community Edition, która oferuje wiele podobnych funkcjonalności.

Jedną z kluczowych cech PyCharm jest jego rozbudowane wsparcie dla języka Python, w tym możliwość wskazywania błędów w czasie rzeczywistym i automatyczne uzupełnianie kodu. Oferuje również bogate funkcje debugowania, takie jak interaktywna konsola, wykresy, punkty przerwania i wiele innych.

PyCharm ma również wiele narzędzi do refaktoryzacji kodu, które pomagają programistom utrzymać kod w czystości i łatwości w utrzymaniu. Narzędzia te obejmują automatyczne formatowanie, przesuwanie i zmienianie nazw zmiennych oraz wiele innych.

Inną ważną cechą PyCharm jest integracja z systemami kontroli wersji, takimi jak Git oraz obsługa wielu frameworków Python.

Podczas tworzenia projektu zostanie użytych dużo dodatkowych bibliotek dla języka Python. Najważniejszą z nich będzie biblioteka Tkinter.

Umożliwia ona programistom tworzenie aplikacji z interfejsem graficznym użytkownika (GUI). Tkinter oferuje wiele wbudowanych metod do tworzenia i stylizacji elementów GUI, takich jak przyciski, etykiety, pola tekstowe i wiele innych. Biblioteka ta pozwala również na dodanie interakcji   
do elementów GUI poprzez mechanizmy obsługi zdarzeń.

Inne wykorzystane biblioteki:

* Pandas - umożliwia wczytywanie, przetwarzanie i zapisywanie danych w różnych formatach, takich jak arkusze kalkulacyjne, pliki CSV i wiele innych. Pandas oferuje wiele funkcji   
  do filtrowania, sortowania i grupowania danych, a także możliwość łączenia danych z różnych źródeł.
* Matplotlib – jest używana do tworzenia wykresów i wizualizacji danych. Umożliwia generowanie różnych rodzajów wykresów, takich jak liniowe, słupkowe, punktowe i wiele innych. Matplotlib oferuje również możliwość dostosowania wyglądu tych wykresów.
* Requests – moduł, który ułatwia wykonywanie zapytań http. Obsługuje on również nagłówki, parametry URL, pliki cookies, uwierzytelnianie i wiele innych funkcji, które są często używane podczas komunikacji z serwerem.

1. **User stories**

Ja jako użytkownik chcę mieć możliwość dodanie do aplikacji materiałów do nauki, aby rozpocząć naukę.

1. Utworzenie możliwości wprowadzania danych do programu poprzez formularz.
2. Utworzenie możliwości wprowadzenia danych do programu za pomocą pliku.

Ja jako użytkownik chcę mieć możliwość zobaczenia raportu po zakończonej sesji nauki, aby posiąść wiedzę odnośnie tego co powinienem powtórzyć.

1. Wygenerowanie raportu na podstawie danych z bieżącej sesji.

Ja jako użytkownik chcę mieć możliwość zresetowania bieżącej sesji nauki, aby rozpocząć naukę od początku.

1. Możliwość zresetowania bieżącej sesji.

Ja jako użytkownik chcę mieć możliwość użycia awaryjnej pauzy podczas sesji nauki, aby zatrzymać chwilowo proces nauki.

1. Możliwość bez konsekwentnego zatrzymania bieżącej sesji za pomocą odpowiedniego przycisku

Ja jako użytkownik chcę mieć możliwość poznania 50 najpopularniejszych słów z danego języka, aby poznać często używane słowa w danym języku.

1. Możliwość nauki najpopularniejszych słów w wybranym języku z menu dostępnego w aplikacji

Ja jako użytkownik chcę mieć możliwość zmiany długości przerw od nauki, aby dostosować proces nauki do swoich wymagań.

1. Możliwość edycji wartości czasu w systemie zliczającym czas nauki.
2. **Narzędzia pomocnicze**



Notion to platforma do organizacji pracy i projektów, która oferuje wiele funkcjonalności, umożliwiających użytkownikom tworzenie spersonalizowanych przestrzeni roboczych i organizowanie swojej pracy w sposób, który odpowiada ich potrzebom. Notion jest dostępny jako aplikacja internetowa, desktopowa oraz aplikacja mobilna dla systemów iOS i Android.

Głównym celem Notion jest umożliwienie użytkownikom tworzenia notatek, list to-do, kalendarzy, tabel, bazy danych, które można łatwo dzielić się z innymi użytkownikami. Notion oferuje również narzędzia do pracy z projektami, takie jak tablice kanban, które pomagają w zarządzaniu projektami i zadaniami.

Program umożliwia programistom integrację z innymi narzędziami pracy, takimi jak Slack, Google Drive, Github i wiele innych. Oferuje również funkcje do importu i eksportu danych,   
które ułatwiają przenoszenie danych między różnymi platformami i narzędziami.

Oferuje on wiele funkcjonalności, które umożliwiają użytkownikom pracę w sposób bardziej produktywny i efektywny. Platforma ta jest wysoce konfigurowalna i dostosowywalna   
do indywidualnych potrzeb użytkowników, co sprawia, że jest bardzo popularna wśród programistów i profesjonalistów w różnych branżach.

1. **Product backlog**
2. Stworzenie repozytorium zdalnego na Github
3. Zaprojektowanie struktury plików aplikacji
4. Zaimplementowanie struktury plików w oparciu o projekt
5. Zaprojektowanie logo aplikacji
6. Zaprojektowanie palety kolorystycznej programu
7. Zaprojektowanie interfejsu głównego aplikacji
8. Zaimplementowanie interfejsu głównego aplikacji w oparciu o projekt
9. Zaimplementowanie funkcjonalności licznika Pomodoro
10. Zaprojektowanie formularza do wprowadzania danych
11. Zaimplementowanie funkcjonalności pobierania danych z przesłanego pliku
12. Zaimplementowane funkcjonalności fiszek
13. Zaimplementowanie funkcji wyświetlającej wprowadzone dane na fiszkach
14. Zaimplementowanie logiki odpowiedzialnej za wyświetlanie fiszek
15. Zaprojektowanie zadania interaktywnego – gra w wisielca
16. Zaprojektowanie zadania interaktywnego – prawda/fałsz
17. Zaprojektowanie zadania interaktywnego – wybierz odpowiednie
18. Zaprojektowanie klasy odpowiedzialnej za zliczanie punktacji
19. Zaimplementowanie klasy odpowiedzialnej za zliczanie punktacji
20. Zaimplementowanie możliwości edycji parametrów licznika Pomodoro
21. Zaimplementowanie klasy odpowiedzialnej za pobieranie najpopularniejszych słów w danym języku
22. Zaimplementowanie funkcjonalności wyświetlającej 50 najpopularniejszych słów w wybranym języku w postaci fiszek
23. Zaimplementowanie funkcji pozwalającej na wstrzymanie czasu licznika Pomodoro