

Dokumentacja do projektu „Fiszki” PIPR

Cel i opis projektu:

Celem projektu było stworzenie elektronicznej wersji fiszek na których umieszcza się słowa w dwóch językach. Projekt zakładał realizację podanych funkcjonalności:

1. definiowanie fiszek
2. importowanie oraz eksportowanie fiszek
3. zarządzanie fiszkami (ustawienie priorytetu, usunięcie)
4. planowanie czasu testów
5. przeprowadzenie testu z predefiniowaną liczbą fiszek
6. wygenerowanie statystyk dotyczących efektów nauki
7. wygenerowanie statystyk dotyczących korzystania z aplikacji

Wszystkie z podanych funkcjonalności zostały zrealizowane i są dostępne w UI.

Działanie:

- Klasa Card:

Obiekt Card() przechowuje informacje na temat każdej fiszki (id, nazwa, nazwa przetłumaczona, kategorie do jakich fiszka należy, częstość wyświetlania).

Obiekt reaguje na odpowiedź na fiszkę i automatycznie zapisuje zmiany na temat samego siebie w bazie danych.

- Klasa CardCollection:

Obiekt CardCollection() przechowuje obiekty Card() które łączy wspólna cecha (kategoria)

Obiekt umożliwia wybór znajdujących się w nim fiszek.

- Klasa Answer:

Obiekt Answer() zawiera informacje o odpowiedzi na fiszkę (odpowiedź podana, odpowiedź poprawna, czy dobra, data odpowiedzi)

- Klasa Exam:

Obiekt Exam() jest tworzony w momencie rozpoczęcia przez użytkownika testu. Przechowuje informacje o pytaniach (fiszkach), odpowiedziach, datę i czy został ukończony. Obiekt może podawać następne pytanie, reagować na odpowiedź na pytanie i generować rezultat w formie do zapisu w celu zachowania statystyk.

- **Klasa Stats:**

Obiekt Stats() działa jako menedżer statystyk. Ładuje statystyki z pliku i je do niego zapisuje. Może zwracać m.in. ilość podjętych testów, wyniki testów, przeszłe odpowiedzi posegregowane według daty, czas użytkowania aplikacji.

Obiekt zapisuje informacje w pliku .json w którym każda statystyka jest odpowiednio sklasyfikowana.

- **utils.py:**

Plik zawiera funkcje związane z łączeniem części logiki z funkcjonalnościami w GUI. Jest to m.in. importowanie i eksportowanie kart

Przechowywanie danych:

Program przechowuje dane w folderze `storage` gdzie znajdują się dwa główne pliki:

- cards.json – przechowujący fiszki dostępne w programie
- stats.json – przechowujący statystyki programu i korzystania z niego

W folderze znajduje się jeszcze folder icons w którym są ikony używane przez GUI

Ułatwienia dostępu:

- stylesheets.py – zawiera klasę StyleSheet zawierającą style używane do dynamicznego stylizowania elementów GUI
- config.py – zawiera klasę Config która posiada stałe wartości używane w celu modyfikacji działania programu. (ścieżka do pliku ze statystykami...)

GUI:

W programie znajdują się dwa pliki odpowiadające za łączenie funkcjonalności z Gui.

gui.py zawiera cały kod głównego okna aplikacji, a card_dialog_gui.py zawiera kod okna dodawania nowej fiszki.

Eksportowanie/Importowanie i dodawanie/usuwanie

- Eksportowanie – Użytkownik wybiera kategorie i po kliknięciu przycisku eksportowania ma do wyboru format .json lub .txt. W obydwu przypadkach utworzą się pliki które będą zawierały listę kart w tej z jej ogólnymi parametrami, czyli bez informacji o tym jak często karta powinna się wyświetlać użytkownikowi.

- Importowanie – Podobnie z poziomu UI użytkownik klika przycisk `Dodaj fiszkę > importuj z pliku`. Następnie do wyboru jest .txt lub .json sformatowany taki sam sposób jak plik który uzyskuje się przy eksportowaniu. Po wybraniu pliku baza fiszek programu zostanie poszerzona o te które jeszcze się w niej nie znajdują (bez dodawania tych samych).
- Dodawać fiszki można ręcznie z poziomu UI gdzie użytkownik ma dodatkowo opcję predefiniowania częstości wyświetlania fiszki.
- Usuwać fiszki można z poziomu UI klikając przycisk usuwania pod aktualnie wyświetlaną fiszką

Program można uruchomić klikając podwójnie plik run.bat albo ręcznie uruchamiając plik gui.py z poziomu głównego katalogu `python .\flashcards\gui_components\gui.py`