

PODSTAWY GRAFIKI KOMPUTEROWEJ

DOKUMENTACJA PROJEKTU

Temat nr 13: Obrazkowe układanki

Kraków, 30 stycznia 2015

Spis treści

1	Obrazkowe układanki	2
2	Założenia wstępne przyjęte w realizacji projektu	2
3	Analiza projektu	2
4	Podział pracy i analiza czasowa	2
4.1	Specyfikacja danych wejściowych	2
4.2	Opis oczekiwanych danych wyjściowych	2
4.3	Zdefiniowanie struktur danych	2
4.4	Specyfikacja interfejsu użytkownika	2
4.5	Wyodrębnienie i zdefiniowanie zadań	2
4.6	Decyzja o wyborze narzędzi programistycznych	2
5	Opracowanie i opis niezbędnych algorytmów	3
6	Kodowanie	3
7	Testowanie	3
8	Wdrożenie,raport i wnioski	3

1 Obrazkowe układanki

Celem projektu było wyświetlenie fragmentu EKG na urządzeniu z systemem Android.

2 Założenia wstępne przyjęte w realizacji projektu

3 Analiza projektu

4 Podział pracy i analiza czasowa

4.1 Specyfikacja danych wejściowych

Dane wejściowe są częściowo dostarczane przez użytkownika, a częściowo ładowane przez program. Po uruchomieniu aplikacja ładuje plik `BestScores.txt` lub tworzy go jeżeli takowy nie istnieje. Zawarte w nim są najlepsze wyniki w zależności od rodzaju gry. Załadowane zostają także obrazki potrzebne w celu uruchomienia gry. Użytkownik wybiera korzystając z interfejsu graficznego obrazek, którego chce użyć do gry oraz opcje gry takie jak rozmiar układanki oraz typ gry.

4.2 Opis oczekiwanych danych wyjściowych

W ramach danych wyjściowych możemy uwzględnić plik `BestScores.txt`, który jest nadpisywany jeżeli użytkownik uzyskał jeden z pięciu najlepszych rezultatów w konkretnym typie rozgrywki, a także wypisanie wyniku gry dla użytkownika.

4.3 Zdefiniowanie struktur danych

Najważniejszą strukturą danych wykorzystaną w programie jest klasa *wxWidget*. Dodatkowo duże znaczenie ma także wrapper dla tej klasy - klasa *Tile*, która udostępnia interfejs, który pomaga w manipulacji poszczególnymi elementami układanki. Najlepsze wyniki zapisywane są w pliku o rozszerzeniu `txt`. W programie do obsługi najlepszych wyników wykorzystywana jest klasa *GameStats* oraz *wxListView* w celu prezentowania ich użytkownikowi.

4.4 Specyfikacja interfejsu użytkownika

Dla użytkownika dostępny jest graficzny interfejs, który ma na celu zapewnić prostotę oraz intuicyjność w użytkowaniu programu. Możliwości kontroli programu oraz wyboru opcji zrealizowane są na podstawie list rozwijanych oraz suwaków dostępnych w bibliotece `WxWidgets`.

4.5 Wyodrębnienie i zdefiniowanie zadań

Realizacja projektu podzielona została na następujące części:

- interfejs graficzny
- obsługa plików
- gra

4.6 Decyzja o wyborze narzędzi programistycznych

Narzędziami programistycznymi wykorzystywanym przez zespół były programy `WxDev` oraz `CodeBlocks`. Pierwszy z nich okazuje się być niestabilnym dlatego częściowo wykorzystywany był także wspomniany `CodeBlocks`. użytymi bibliotekami były: `WxWidget` oraz biblioteka standardowa języka `c++` w wersji `C11`. Dodatkowo zespół korzystał z systemu kontroli wersji `GIT`, który wraz z wykorzystaniem serwisu internetowego `GitHub` skutecznie wspomaga realizację tego typu projektów.

5 Opracowanie i opis niezbędnych algorytmów

6 Kodowanie

7 Testowanie

Testowanie odbywało się równolegle z tworzeniem aplikacji. Poszczególne klasy zostały przetestowane przez osoby odpowiedzialne za ich tworzenie. Po fazie implementacji przeprowadzone zostały testy manualne, które przebiegły pomyślnie.

8 Wdrożenie,raport i wnioski

Aplikacja została napisana dla systemu operacyjnego Windows, gdzie uruchamia się i działa poprawnie bez dodatkowy bibliotek czy komponentów, które są załączone bezpośrednio w pliku exe. Wszystkie założenia projektowe zostały spełnione. Ewentualny rozwój mógłby zakładać rozbudowanie aplikacji o możliwość wybrania obrazka przez użytkownika oprócz możliwości wybrania jednego z domyślnych, poprawienie estetyki interfejsu graficznego oraz wprowadzenia płynnych przejść przy układaniu obrazka.