



AKADEMIA GÓRNICZO-HUTNICZA

Dokumentacja do projektu

tytuł projektu

z przedmiotu

Języki programowania obiektowego

tu wpisać kierunek i rok studiów

imię i nazwisko wykonawcy(ów)

tutaj wpisać grupę np. poniedziałek 14:40

prowadzący: Rafał Frączek

tutaj wpisać datę wykonania dokumentacji

1. Opis projektu

W tym punkcie należy umieścić ogólny opis projektu wraz z informacją o tym do czego on służy.

2. Project description

Here the project description in English must be provided.

3. Instrukcja użytkownika

W tym punkcie należy umieścić instrukcję użytkowania programu. Może być to na przykład opis poszczególnych menu w programie. W przypadku gry należy opisać zasady gry. Opcjonalnie można wstawić zrzuty ekranu. Jeśli uruchomienie programu wymaga wykonania jakiś niestandardowych lub dodatkowych czynności (na przykład uruchomienie serwera baz danych itp.) to należy zamieścić tę informację.

4. Kompilacja

W tym punkcie podajemy informację o kompilacji projektu. Podajemy informację o tym czy wystarczy standardowa kompilacja czy też konieczne jest wykonanie niestandardowej kompilacji. Jeśli projekt działa tylko w jednym systemie operacyjnym (np. Linux) to podajemy informację o tym fakcie.

5. Pliki źródłowe

W tym punkcie należy opisać wszystkie pliki źródłowe (.cpp, .h) w projekcie. Należy podać nazwę każdego pliku oraz informację o tym co się w nim znajduje. Na przykład:

Projekt składa się z następujących plików źródłowych:

- *BankAccount.h*, *BankAccount.cpp* – deklaracja oraz implementacja klasy *BankAccount*,
- *Globals.h*, *Globals.cpp* – deklaracja oraz instancje zmiennych globalnych w projekcie,
- *Utils.h*, *Utils.cpp* – deklaracja oraz implementacja różnych przydatnych funkcji.

6. Zależności

Jeśli projekt wykorzystuje jakieś dodatkowe zewnętrzne biblioteki (oprócz biblioteki standardowej STL) to podajemy listę użytych dodatkowych bibliotek wraz z krótkim opisem oraz linkiem do strony internetowej. Jeśli projekt nie wykorzystuje zewnętrznych bibliotek to wpisujemy słowo „brak”. Na przykład:

W projekcie wykorzystano następujące dodatkowe biblioteki:

- SQLite – silnik bazodanowy SQL. Strona internetowa: <https://sqlite.org/>.
- TCLAP – biblioteka szablonów C++ do parsowania argumentów podanych w linii poleceń. Strona internetowa: <http://tclap.sourceforge.net/>.

7. Opis klas

W tym punkcie należy umieścić opis wszystkich stworzonych w projekcie klas. Należy podać do czego służy dana klasa oraz informację o jej publicznych metodach. Opcjonalnie można załączyć fragmenty kodu źródłowego. Na przykład:

W projekcie utworzono następujące klasy:

- *BankAccount* – reprezentuje konto w systemie bankowym.
 - `float get_balance(void)` – zwraca bieżący stan konta,

- `void withdraw(float ammount)` – pobiera z konta sumę `ammount`,
- `Globals` – klasa zawierająca publiczne oraz statyczne metody i zmienne.
 - `void save(const string& file_path)` – zapisuje bieżący stan programu,
 - `int num_accounts` – całkowita liczba kont w systemie.
- `Utils` – klasa zawierająca różne przydatne funkcje.
 - `bool contains(const string& str, const string& key)` – zwraca `true` gdy `str` zawiera `key`.
 - `vector<string> split(const string& str, const string& delim)` – dzieli `str` na składniki rozdzielone przez `delim`.

8. Zasoby

Jeśli projekt wykorzystuje jakieś dodatkowe zasoby jak na przykład pliki z danymi tekstowymi, pliki obrazów itp. to w tym punkcie należy je wyszczególnić. W przypadku plików tekstowych konieczne jest opisanie struktury takiego pliku. Jeśli w projekcie nie ma żadnych zasobów to piszemy słowo „brak”. Na przykład:

W projekcie wykorzystywane są następujące pliki zasobów:

- `customers.txt` – plik zawierający dane klientów banku. Struktura pliku:
 - pierwsza linia: imię i nazwisko klienta,
 - druga linia: PESEL,
 - trzecia linia: numer konta w banku,
 - w następnych liniach dane kolejnych klientów w powyższej kolejności.
- `transactions.db` – plik binarny bazy danych SQLite zawierający wykonane transakcje bankowe.

9. Dalszy rozwój i ulepszenia

W tym punkcie podajemy możliwe dalsze kierunki rozwoju projektu. Można podać na przykład kolejne funkcjonalności, które mogłyby być zaimplementowane w projekcie.

10. Inne

W tym punkcie podajemy opcjonalnie inne informacje nieuwzględnione w poprzednich punktach. W przypadku braku dodatkowych informacji wpisujemy słowo „brak”.