

AKADEMIA NAUK STOSOWANYCH W NOWYM SĄCZU

Wydział Nauk Inżynieryjnych
Katedra Informatyki

DOKUMENTACJA PROJEKTOWA PROGRAMOWANIE URZĄDZEŃ MOBILNYCH

Aplikacja Survivalowa

Autor:
Bogusław Gruca
Karol Kowalik

Prowadzący:
mgr inż. Dawid Kotlarski

Nowy Sącz 2022

Spis treści

1. Ogólne określenie wymagań	3
1.1. Przykład	3
1.2. Instalacja	5
2. Określenie wymagań szczegółowych	6
3. Projektowanie	7
4. Implementacja	8
5. Testowanie	9
6. Podręcznik użytkownika	10
Literatura	11
Spis rysunków	11
Spis tabel	12
Spis listingów	13

1. Ogólne określenie wymagań

Celem programu jest utworzenie prostej w obsłudze oraz uniwersalnej w każdej sytuacji aplikacji, dzięki której będzie możliwe bezpieczne poruszanie się w terenie. Aplikacja powinna zawierać wszystkie najważniejsze narzędzia do posługiwania się poza domem. Odbiorcą ma być każdy, niezależnie od wieku liczy się prostota w obsłudze, aby nikt nie miał problemu z użytkowaniem. Aplikacja ma być obsługiwana przez system Android. Program musi posiadać dostęp do mapy, która będzie umożliwiała orientowanie się w terenie górzystym, zabudowanym, leśnym.

1.1. Przykład

Tak zaczynamy pisanie pierwszego akapitu. Jeśli chcemy napisać przypis do bibliografii wykonujemy to w ten sposób¹.



Rys. 1.1. Logo

Tutaj może coś być wpisane.

Tutaj może coś być wpisane². Rysunek 1.1 (s. 3) pokazuje przykładową ilustrację.

Tab. 1.1. Tabelka przykładowa

U_n	I_{zw}
kV	%
7.2	100

Listing kodu

```
1 #include <iostream>
2 #include <cstdlib>
3 #include <ctime>
4 using namespace std;
5
6 /*
```

¹Przykład odnośnika do książki[legierski].

²Przykład odnośnika do strony www[www1].

```
7  liczby pseldolosowe
8  */
9
10 int main(int argc, char** argv) {
11
12     int tab[10][10];
13
14     for(int i=0;i<10;i++)
15     for(int j=0;j<10;j++)
16     tab[i][j]=0;
17
18     srand(time(NULL));    //generowanie z czasu
19     int min=3;
20     int max=7;
21     for(int i=0;i<10;i++)
22     for(int j=0;j<10;j++)
23     tab[i][j]=(rand()%(max-min+1))+min;
24
25     for(int i=0;i<10;i++)
26     {
27         for(int j=0;j<10;j++)
28             cout<<tab[i][j]<<" ";
29         cout<<endl;
30     }
31
32     return 0;
33 }
```

Listing 1. Przykładowy kod 001

Tutaj może coś być wpisane. Tutaj może coś być wpisane. Tutaj może coś być wpisane. Tabela 1.1 (s. 3) pokazuje sposoby użycia trybu matematycznego.

Kod 1 (s. 3) przedstawia sposób generowania liczb pseudolosowych. Kod 2 (s. 4) przedstawia generowanie pliku HTML.

Alternatywna metoda wklejenia kodu:

```
1 #include <iostream>
2 #include <fstream>
3 using namespace std;
4
5 int main(int argc, char** argv) {
6
7     ofstream plik("strona.html");
8     if(!plik)
9         cout<<"blad zapisu pliku";
```

```

10  else
11      {
12          plik<<"<html>";
13          plik<<"<head><title>Moja pierwsza strona www</title></head>";
14          plik<<"<body>Strona WWW</body>";
15          plik<<"</html>";
16
17          cout<<"Wygenerowana strona";
18      }
19      plik.close();
20
21
22      return 0;
23 }

```

Listing 2. Przykładowy kod 002

1.2. Instalacja

Poniżej są opisane kroki potrzebne do instalacji L^AT_EX'a oraz do używania tego szablonu.

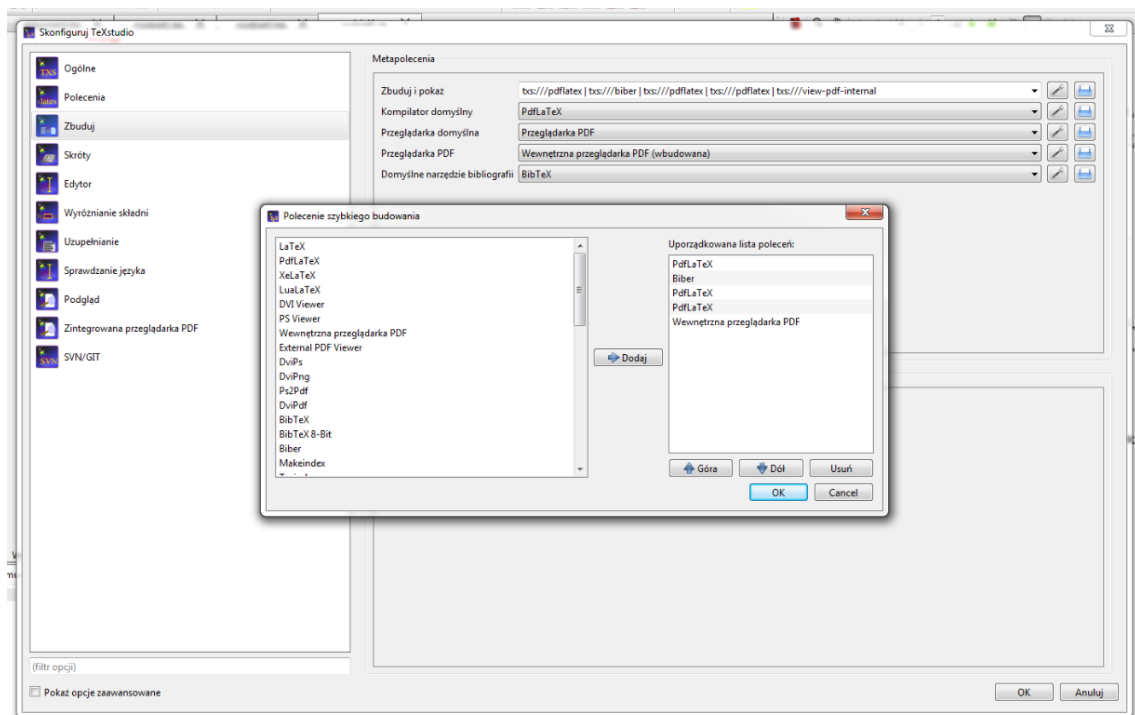
Na początku instalujemy T_EXLive³. Ściągamy plik instalacyjny, zajmuje około 25MB. Podczas instalacji można wybrać do zainstalowania różne kolekcje pakietów. Jeśli nie ma problemów z miejscem na dysku to można zainstalować wszystkie, wtedy nie będzie problemu z brakującymi pakietami i błędami. Po wybraniu kolekcji brakujące pliki są pobierane z internetu. Pełna instalacja programu zajmuje około 8GB. Najlepiej zostawić instalację na noc, ponieważ proces zabiera sporo czasu. Warto ustawić komputer tak, aby się nie wyłączył lub nie uśpił. Warto także przed instalacją zablokować antywirusa, ponieważ może blokować niektóre z komponentów.

Następnie instalujemy T_EXstudio⁴. Ściągamy plik instalacyjny zajmujący około 120MB. Instalacja przebiega standardowo.

Następnym krokiem jest ustawienie w T_EXstudio kolejności budowania projektu. Należy wybrać zakładkę: „Opcje/Konfiguruj T_EXstudio...”. W otwartym oknie przechodzimy na zakładkę „Zbuduj”. Na rysunku 1.2 (s. 6) pokazany jest zrzut ekranu z konfiguracją. W linijce „Zbuduj i pokaż” klikamy ikonę klucza, żeby przejść do konfiguracji polecenia. W otwartym oknie ustawić kolejność tak jak pokazano na rysunku.

³Instalka na stronie <https://www.tug.org/texlive/acquire-netinstall.html>[**www2**].

⁴Plik instalacyjny na stronie <https://www.texstudio.org>[**www3**].



Rys. 1.2. Ustawienie TeXstudio

2. Określenie wymagań szczegółowych

Naszym celem w budowaniu aplikacji jest zaprogramowanie tak, aby posiadała następujące funkcje:

- mapę
- latarke
- kompas
- stoper który będzie odliczał czas do przerwy
- lokalizacja

Aplikacja będzie miała prosty interfejs. Brak internetu będzie skutkował utrudnieniami w korzystaniu z aplikacji. Zaczniemy budowę programu od zaprojektowania graficznego aplikacji i ułożenia danych narzędzi, przycisków. Po ustaleniu grafiki i przycisków funkcyjnych, przejdziemy do rozpoczęcia kodowania aplikacji i podłączać mapę z lokalizacją. Po zaprogramowaniu aplikacji będziemy sprawdzali jej prawidłowe działanie i poprawiali błędy. Program będziemy rozwijali w przyszłości o kolejne funkcje.

3. Projektowanie

4. Implementacja

5. Testowanie

6. Podręcznik użytkownika

Spis rysunków

1.1. Logo	3
1.2. Ustawienie TeXstudio	6

Spis tabel

1.1. Tabelka przykładowa	3
------------------------------------	---

Spis listingów

1.	Przykładowy kod 001	3
2.	Przykładowy kod 002	4