|  |
| --- |
| HiveHub |
| Aplikacja wspomagająca i zarządzająca zbiorem osób |
| Jakub Rudnicki – Studia Stacjonarne I stopnia |

# Spis treści

[1 Spis treści 1](#_Toc156473039)

[2 Opis ogólny HiveHub 3](#_Toc156473040)

[2.1 Sformułowanie zadania 3](#_Toc156473041)

[2.1.1 Treść Zadania 3](#_Toc156473042)

[2.1.2 Interpretacja zadania 3](#_Toc156473043)

[2.1.3 Problematyka projektu 3](#_Toc156473044)

[2.1.4 Struktura Danych 4](#_Toc156473045)

[2.1.4.1 Opis pól struktury 4](#_Toc156473046)

[2.2 Zadania realizowane przez HiveHub 4](#_Toc156473047)

[3 Schematy blokowe 5](#_Toc156473048)

[3.1 Schemat Blokowy przedstawiający funkcje main 5](#_Toc156473049)

[3.2 Schemat blokowy przedstawiający działanie funkcji dodawania użytkownika do bazy danych („userCreator”) 6](#_Toc156473050)

[3.3 Wykorzystane oprogramowanie do tworzenia schematów blokowych i dodatkowe informacje 7](#_Toc156473051)

[3.3.1 Oprogramowanie 7](#_Toc156473052)

[3.3.2 Informacje dodatkowe 7](#_Toc156473053)

[4 Schemat plików 8](#_Toc156473054)

[4.1 Opis struktury katalogów 9](#_Toc156473055)

[4.1.1 App 9](#_Toc156473056)

[4.1.2 App/HiveHub 9](#_Toc156473057)

[4.1.2.1 Config: Katalog zawierający pliki związane z konfiguracją aplikacji. 9](#_Toc156473058)

[4.1.2.2 Database: Katalog zawierający pliki związane z obsługą bazy danych. 9](#_Toc156473059)

[4.1.2.3 Methods: Katalog zawierający pliki związane z funkcjami i metodami aplikacji. 9](#_Toc156473060)

[4.1.2.4 GUI\_Features: Katalog z funkcjami interfejsu graficznego. 9](#_Toc156473061)

[4.1.2.5 User\_Display\_Features: Katalog z funkcjami wyświetlania użytkowników. 9](#_Toc156473062)

[4.1.3 Doc 10](#_Toc156473063)

[4.2 Platforma i Wymagania Systemowe 10](#_Toc156473064)

[4.2.1 Język i Narzędzia 10](#_Toc156473065)

[4.2.2 System Operacyjny 10](#_Toc156473066)

[4.2.3 Wymagania Sprzętowe 10](#_Toc156473067)

[4.2.4 Zalecenia 10](#_Toc156473068)

[4.3 Możliwe zastosowanie aplikacji 10](#_Toc156473069)

[5 Instrukcje Instalacji HiveHub 11](#_Toc156473070)

[5.1 Wymagania Przed instalacyjne: 11](#_Toc156473071)

[5.2 Kroki Instalacji: 11](#_Toc156473072)

[5.2.1 Pobranie Kodu Źródłowego: 11](#_Toc156473073)

[5.2.2 Przygotowanie Środowiska: 11](#_Toc156473074)

[5.2.3 Kompilacja za pomocą CMake: 12](#_Toc156473075)

[5.2.4 Uruchomienie Aplikacji: 12](#_Toc156473076)

[5.2.5 Uruchomienie Aplikacji w Konsoli: 12](#_Toc156473077)

[5.2.6 Finalizacja instalacji 12](#_Toc156473078)

[6 Przykłady działania oprogramowania 13](#_Toc156473079)

[6.1 Pierwsze uruchomienie aplikacji 13](#_Toc156473080)

[6.2 Dodawanie nowego użytkownika 15](#_Toc156473081)

[6.3 Edycja danych – zmiana imienia 19](#_Toc156473082)

[6.4 Edycja Danych – zmiana nazwiska 22](#_Toc156473083)

[6.5 Edycja danych – zmiana zainteresowań 24](#_Toc156473084)

[6.6 Edycja danych – usuwanie użytkownika 26](#_Toc156473085)

[6.7 Wyświetlanie użytkowników po email 28](#_Toc156473086)

[6.8 Wyświetlanie użytkowników po zainteresowaniach 31](#_Toc156473087)

[6.9 Wyświetlanie wszystkich użytkowników 33](#_Toc156473088)

[6.10 Wyjście z aplikacji 34](#_Toc156473089)

[7 Źródła 35](#_Toc156473090)

# Opis ogólny HiveHub

*HiveHub* to aplikacja napisana w języku C++, wykorzystująca CMake do budowy i zarządzania projektem. Jej głównym celem jest wspomaganie i zarządzanie zbiorem osób. Aplikacja umożliwia użytkownikom dodawanie nowych osób do zbioru, edycję informacji takich jak imię, nazwisko czy zainteresowania, a także usuwanie osób z tego zbioru. W skrócie, HiveHub służy do efektywnego gromadzenia i zarządzania informacjami o różnych jednostkach.

## Sformułowanie zadania

### Treść Zadania

„Projekt i implementacji systemu wspomagającego zarządzanie zbiorem adresów osób. Automatyczne grupowanie po zainteresowaniach”

### Interpretacja zadania

Stworzenie aplikacji umożliwiającego wyświetlanie zbioru osób po podanych zainteresowaniach. Aplikacja typu książka adresowa, umożliwiająca dodawanie osób do bazy danych, usuwanie osób, zmianę danych z możliwością wyszukania po kluczu bazy danych bądź wyświetlenie wszystkich rekordów.

### Problematyka projektu

Projekt zakłada stworzenie aplikacji o nazwie HiveHub, której głównym zadaniem jest wspomaganie zarządzania zbiorem adresów osób poprzez automatyczne grupowanie ich według zainteresowań. Aplikacja ma charakter książki adresowej, umożliwiającej dodawanie, usuwanie oraz edycję informacji o osobach w bazie danych. Kluczowym elementem jest również efektywne wyszukiwanie danych przy użyciu klucza bazy danych oraz możliwość wyświetlania wszystkich rekordów.

Projekt napotyka na szereg istotnych problemów, które wymagają szczegółowego rozwiązania:

* Zbieranie danych o osobach:

Zadanie to jest pracochłonne i narażone na błędy, co może prowadzić do niespójności w kwestii segregacji danych. HiveHub ma na celu zastosowanie odpowiednich algorytmów i funkcji, aby usprawnić proces zbierania danych, eliminując błędy i przyspieszając operacje.

* Skomplikowane operacje na danych:

Zarządzanie zbiorem osób wymaga elastycznych mechanizmów dodawania, edycji i usuwania informacji. Wyzwaniem jest zapewnienie, aby te operacje były intuicyjne dla użytkowników, a jednocześnie umożliwiały dokładne zarządzanie informacjami.

* Automatyczne grupowanie po zainteresowaniach:

Projekt stawia sobie za zadanie automatyczne grupowanie osób na podstawie ich zainteresowań. To wyzwanie obejmuje stworzenie skutecznych algorytmów grupowania, uwzględniających różnorodność i zmienność zainteresowań użytkowników.

* Intuicyjność interfejsu użytkownika:

Aplikacja musi charakteryzować się łatwością obsługi, nawet dla użytkowników bez doświadczenia w zarządzaniu danymi. Stworzenie intuicyjnego interfejsu użytkownika jest zatem kluczowe dla sukcesu projektu.

* Wykorzystanie w różnych kontekstach:

HiveHub ma potencjał znalezienia zastosowania w różnych dziedzinach. Wyzwaniem jest projektowanie systemu wystarczająco elastycznego, aby był użyteczny w kontekstach biznesowych, edukacyjnych i naukowych.

### Struktura Danych

Aby efektywnie zarządzać danymi o osobach, projekt wykorzystuje specjalnie zaprojektowaną strukturę danych o nazwie `UserData`. Jest to kluczowy element systemu, który umożliwia przechowywanie, manipulację oraz dostęp do informacji o użytkownikach. Poniżej znajduje się szczegółowa definicja tej struktury:

struct UserData {

string email;

string firstName;

string lastName;

string address;

vector<string> interests;

};

#### Opis pól struktury

* email: Reprezentuje adres e-mail użytkownika. Każdy użytkownik w systemie identyfikowany jest unikalnym adresem e-mail, co zapewnia jednoznaczną identyfikację.
* firstName: Zawiera imię użytkownika.
* lastName: Zawiera nazwisko użytkownika.
* address: Przechowuje adres zamieszkania użytkownika.
* interests: Jest to wektor stringów, który reprezentuje zainteresowania danego użytkownika. Dzięki temu pole, system umożliwia dynamiczne przechowywanie różnorodnych zainteresowań.

Struktura UserData została starannie zaprojektowana tak, aby umożliwiała elastyczne i jednocześnie zorganizowane przechowywanie informacji o osobach. Jej zastosowanie w systemie HiveHub pozwala na spójne gromadzenie danych oraz wygodne operowanie nimi w trakcie dodawania, edycji czy usuwania rekordów z bazy danych.

Wprowadzenie struktury UserData w ramach projektu to kluczowy element, który wspiera zrozumienie struktury danych używanej przez aplikację. To także zapewnia spójność i czytelność kodu, ułatwiając programistom pracę z danymi o użytkownikach.

## Zadania realizowane przez HiveHub

1. Dodawanie Osób: HiveHub umożliwia dodawanie nowych osób do zbioru, zapewniając elastyczność w definiowaniu informacji o każdej z nich.
2. Edycja Informacji: Użytkownicy mają możliwość edytowania danych osób w zbiorze, takich jak imię, nazwisko czy zainteresowania, co pozwala na bieżące aktualizowanie informacji.
3. Usuwanie Osób: Aplikacja umożliwia wygodne usuwanie osób z zbioru w przypadku potrzeby aktualizacji bądź oczyszczenia danych.
4. Filtrowanie i Wyświetlanie: HiveHub pozwala na wyświetlanie osób spełniających określone kryteria, takie jak zainteresowania czy konkretny adres email. Użytkownicy mogą dostosować widok zbioru według własnych potrzeb.

# Schematy blokowe

## Schemat Blokowy przedstawiający funkcje main

## Schemat blokowy przedstawiający działanie funkcji dodawania użytkownika do bazy danych („userCreator”)

## Wykorzystane oprogramowanie do tworzenia schematów blokowych i dodatkowe informacje

### Oprogramowanie

Schematy blokowe zostały w pełni wykonane przy użyciu oprogramowania Draw.io. Draw.io to zaawansowane narzędzie do tworzenia diagramów online, charakteryzujące się prostotą obsługi oraz szerokimi możliwościami personalizacji. Działa w przeglądarkach internetowych, co pozwala na łatwe udostępnianie, współpracę i edycję projektów zespołowych.

Oficjalna strona oprogramowania:

[https://app.diagrams.net/](https://app.diagrams.net/:)

### Informacje dodatkowe

Dodatkowe ciemne wersje schematów blokowych jak i pliki umożliwiające edycje znajdują się w projekcie w katalogu /Doc/Block\_Schemes.

# Schemat plików

Obraz zawierający tekst, zrzut ekranu, menu, Czcionka

Opis wygenerowany automatycznie

## Opis struktury katalogów

### App

* CMakeLists.txt: Plik konfiguracyjny dla CMake, który zawiera instrukcje budowy projektu w katalogu "App".
* CMakePresets.json: Plik konfiguracyjny dla CMake, zawierający predefiniowane zestawy opcji kompilacji.
* LICENSE: Plik zawierający informacje o licencji projektu.

### App/HiveHub

* CMakeLists.txt: Plik konfiguracyjny dla CMake, który zawiera instrukcje budowy projektu w katalogu "HiveHub".
* methods.h: Plik nagłówkowy z deklaracjami ogólnych metod.
* miscellaneous\_functions.cpp: Plik źródłowy z różnymi funkcjami pomocniczymi.
* person\_addition.cpp: Plik źródłowy z funkcją dodawania osoby.
* common\_includes.h: Plik nagłówkowy zawierający wspólne nagłówki dla całego projektu.
* main.cpp: Główny plik źródłowy programu.
* main.h: Plik nagłówkowy głównego pliku programu.

#### Config: Katalog zawierający pliki związane z konfiguracją aplikacji.

* config.h: Plik nagłówkowy z ustawieniami konfiguracyjnymi.
* config\_manager.cpp: Plik źródłowy odpowiadający za zarządzanie konfiguracją.
* console\_utils.h: Plik nagłówkowy z narzędziami do obsługi konsoli.
* database\_setup.cpp: Plik źródłowy zawierający logikę ustawiania bazy danych.
* set\_console.cpp: Plik źródłowy z funkcjami ustawiania konsoli.

#### Database: Katalog zawierający pliki związane z obsługą bazy danych.

* database\_controller.h: Plik nagłówkowy z kontrolerem bazy danych.
* database\_read\_logic.cpp: Plik źródłowy z logiką odczytu z bazy danych.
* database\_save\_logic.cpp: Plik źródłowy z logiką zapisu do bazy danych.
* line\_parser.cpp: Plik źródłowy z parserem linii.

#### Methods: Katalog zawierający pliki związane z funkcjami i metodami aplikacji.

* Edits\_Features: Katalog z funkcjami edycji.
* edit\_first\_name.cpp: Plik źródłowy z funkcją edycji imienia.
* edit\_interests.cpp: Plik źródłowy z funkcją edycji zainteresowań.
* edit\_last\_name.cpp: Plik źródłowy z funkcją edycji nazwiska.
* edit\_methods.h: Plik nagłówkowy z ogólnymi metodami edycji.
* remove\_person.cpp: Plik źródłowy z funkcją usuwania osoby.

#### GUI\_Features: Katalog z funkcjami interfejsu graficznego.

* main\_menu\_interface.cpp: Plik źródłowy z interfejsem głównego menu.
* user\_display\_tool.cpp: Plik źródłowy z interfejsem narzędzia wyświetlania użytkowników.
* user\_edit\_tool.cpp: Plik źródłowy z interfejsem narzędzia edycji użytkowników.
* user\_interface.h: Plik nagłówkowy z interfejsem użytkownika.

#### User\_Display\_Features: Katalog z funkcjami wyświetlania użytkowników.

* display\_all\_users.cpp: Plik źródłowy z funkcją wyświetlania wszystkich użytkowników.
* display\_by\_email.cpp: Plik źródłowy z funkcją wyświetlania użytkownika po adresie e-mail.
* display\_by\_interests.cpp: Plik źródłowy z funkcją wyświetlania użytkowników według zainteresowań.
* display\_features.h: Plik nagłówkowy z ogólnymi funkcjami wyświetlania.

### Doc

* Block\_Schemes: Katalog zawierający schematy blokowe projektu.
* Documents: Katalog zawierający dokumentację projektu.

## Platforma i Wymagania Systemowe

### Język i Narzędzia

Aplikacja została napisana w języku C++ i korzysta z narzędzia CMake do zarządzania projektem.

### System Operacyjny

* HiveHub jest kompatybilny z systemem Windows 7 bądź nowszym.
* HiveHub działa także z rodziną systemów Linux po odpowiedniej edycji plików CMake.

### Wymagania Sprzętowe

* Pamięć RAM: Minimum 20MB.
* Przestrzeń Dyskowa: Około 60MB wolnej przestrzeni dyskowej.
* Karta Graficzna: Aplikacja działa w trybie konsolowym, więc nie wymaga karty graficznej.

### Zalecenia

* Aby korzystać z aplikacji, zalecane jest posiadanie środowiska umożliwiającego obsługę interfejsu wiersza poleceń (CLI)
* Aplikacja nie wymaga dostępu do sieci, z wyjątkiem sytuacji, gdy konieczne jest pobranie repozytorium projektu.

## Możliwe zastosowanie aplikacji

Aplikacja HiveHub może znaleźć zastosowanie w różnych kontekstach, gdzie istnieje potrzeba efektywnego gromadzenia i zarządzania informacjami o osobach, a także automatycznego grupowania tych osób po zainteresowaniach.

* Bazy Danych Klientów:
  + W firmach, które utrzymują bazy danych swoich klientów, HiveHub może być wykorzystywany do skutecznego zarządzania informacjami o klientach, umożliwiając personalizację interakcji i lepsze zrozumienie ich preferencji.
* Zarządzanie Zespołem:
  + W środowiskach biznesowych HiveHub może pełnić rolę narzędzia do zarządzania danymi pracowników, ułatwiając dodawanie nowych osób, aktualizację informacji oraz automatyczne grupowanie pracowników o podobnych zainteresowaniach.
* Organizacja Wydarzeń:
  + Przy organizacji konferencji, szkoleń czy innych wydarzeń, HiveHub może pomóc w skutecznym zarządzaniu informacjami o uczestnikach, umożliwiając szybkie dostosowanie oferty do ich zainteresowań.
* Badania Marketingowe:
  + W dziedzinie marketingu HiveHub może być stosowany do analizy preferencji klientów, co pozwala na lepsze dostosowanie strategii marketingowej do oczekiwań grupy docelowej.
* Bazy Danych Studentów na Uczelni:
  + Na uczelniach aplikacja może być używana do zarządzania danymi studentów, ułatwiając administrowanie informacjami o kursach, zainteresowaniach i historii edukacyjnej.

# Instrukcje Instalacji HiveHub

Aby zainstalować HiveHub na systemie Windows, postępuj zgodnie z poniższymi krokami:

## Wymagania Przed instalacyjne:

1. System Operacyjny:

* Upewnij się, że korzystasz z systemu operacyjnego Windows.

1. Pamięć RAM:

* Sprawdź, czy masz co najmniej 20MB dostępnej pamięci RAM.

1. Przestrzeń Dyskowa:

* Upewnij się, że na dysku jest dostępne około 60MB wolnej przestrzeni dyskowej.

1. Dostęp do Sieci:

* Upewnij się, że posiadasz dostęp do sieci.

## Kroki Instalacji:

### Pobranie Kodu Źródłowego:

* Sklonuj repozytorium projektu ze strony GitHub lub pobierz archiwum ZIP i rozpakuj je na swoim komputerze.

<https://github.com/HyperekDerg/HiveHub.git>

### Przygotowanie Środowiska:

* Przed przystąpieniem do kompilacji projektu HiveHub w języku C++, upewnij się, że Twoje środowisko programistyczne jest odpowiednio skonfigurowane. Poniżej znajdziesz szczegółowe kroki przygotowawcze:
  + Upewnij się, że na Twoim systemie zainstalowany jest odpowiedni kompilator obsługujący standard C++. Dla systemu Windows zaleca się korzystanie z kompilatora MinGW lub Visual Studio C++.
  + Sprawdź, czy masz zainstalowane narzędzie CMake. Jeśli nie, pobierz i zainstaluj najnowszą wersję z oficjalnej strony internetowej CMake

<https://cmake.org/download/>.

* + Wybierz edytor tekstu lub środowisko programistyczne, w którym czujesz się komfortowo. Popularne opcje obejmują Visual Studio Code, Visual Studio, Code::Blocks lub inne, zależnie od preferencji.
  + Jeśli nie masz jeszcze systemu kontroli wersji, zainstaluj narzędzie takie jak Git. Zapewni to łatwiejszą pracę nad projektem, umożliwiając śledzenie zmian i przywracanie poprzednich wersji kodu.
* Po wykonaniu tych kroków, Twoje środowisko powinno być gotowe do kompilacji i uruchomienia projektu HiveHub. Przed przystąpieniem do procesu kompilacji, zaleca się dokładne zapoznanie się z dokumentacją projektu, aby uniknąć ewentualnych problemów i zastosować się do ewentualnych dodatkowych zaleceń dotyczących konfiguracji środowiska.

### Kompilacja za pomocą CMake:

1. Otwórz terminal w katalogu głównym projektu.
2. Utwórz folder 'build': mkdir build
3. Przejdź do folderu 'build': cd build
4. Skonfiguruj projekt za pomocą CMake: cmake ..
5. Skompiluj projekt: cmake --build .

### Uruchomienie Aplikacji:

* Po skompilowaniu projektu, znajdź plik wykonywalny (np. HiveHub.exe) w katalogu wynikowym.

### Uruchomienie Aplikacji w Konsoli:

* Uruchom terminal w miejscu, gdzie znajduje się plik wykonywalny.
* Wprowadź komendę: ./HiveHub.exe

### Finalizacja instalacji

* Teraz możesz korzystać z HiveHub, dodawać, edytować, usuwać osoby oraz przeglądać zbiór według różnych kryteriów.
* Przy pierwszym uruchomieniu aplikacji rozpocznie się proces konfiguracji. Jeżeli chcesz dowiedzieć się więcej to sprawdź w dokumentacji sekcję *„Przykłady działania programu”.*

# Przykłady działania oprogramowania

## Pierwsze uruchomienie aplikacji

Obraz zawierający tekst, zrzut ekranu, design

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Dodawanie nowego użytkownika

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

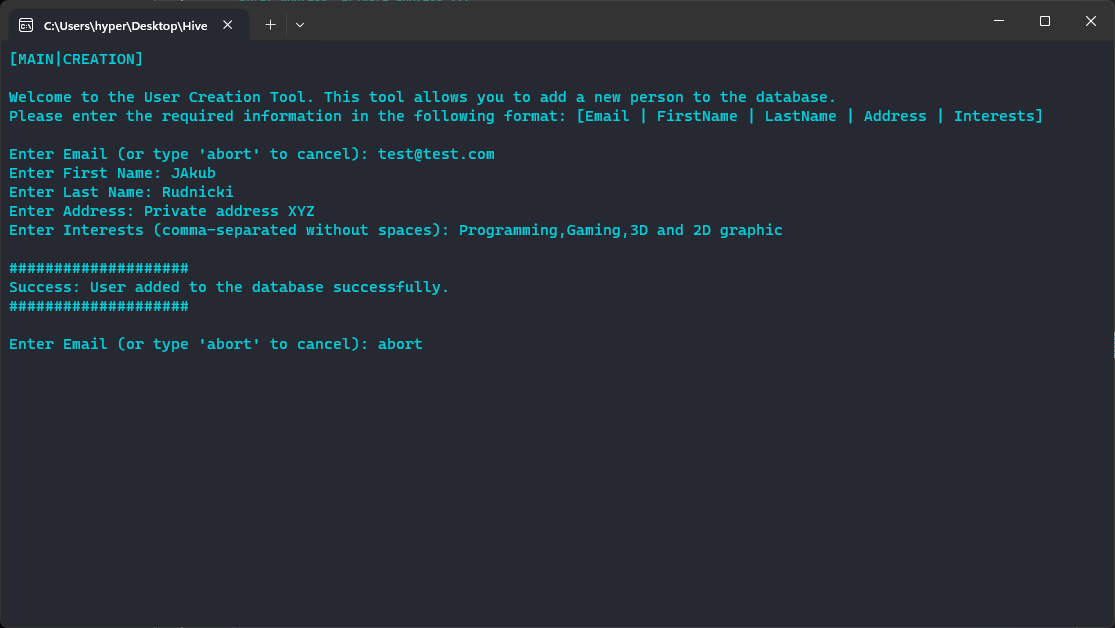
Opis wygenerowany automatycznie

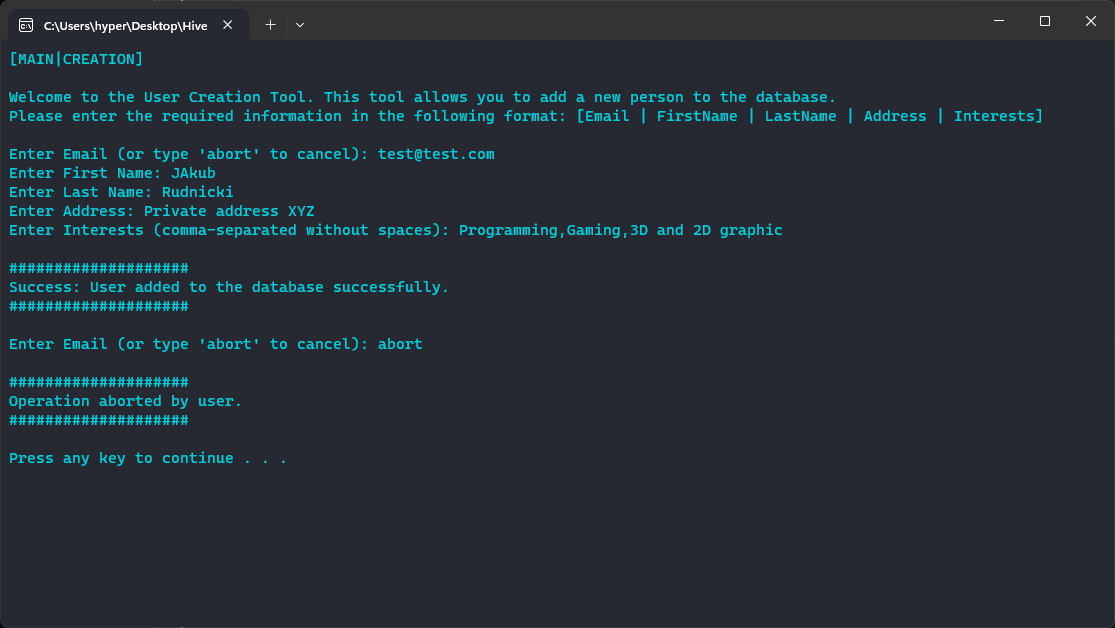
Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie





Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Edycja danych – zmiana imienia

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Edycja Danych – zmiana nazwiska

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Edycja danych – zmiana zainteresowań

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Edycja danych – usuwanie użytkownika

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Wyświetlanie użytkowników po email

Uwaga poprzednio użytkownik [test@test.com](mailto:test@test.com) został usunięty. Dlatego dodano nowego użytkownika testowego [drabina@test.com](mailto:drabina@test.com) i [poziomka@test.com](mailto:poziomka@test.com)

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Wyświetlanie użytkowników po zainteresowaniach

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

## Wyświetlanie wszystkich użytkowników

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

## Wyjście z aplikacji

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

# Źródła

* <https://www.zenflowchart.com/guides/switch-case-flowchart>
* <https://www.zenflowchart.com/flowchart-symbols>
* <https://www.w3schools.com/cpp/default.asp>
* <https://mirek.ii.uph.edu.pl/programowanie/>
* <https://cmake.org/cmake/help/latest/manual/cmake.1.html>
* <https://www.drawio.com/doc/>