Generator sztucznych danych osobowych

Ewa Pasterz

Grudzień 2022

Spis treści

1 Opis projektu

Celem projektu jest stworzenie aplikacji generującej sztuczne dane osobowe. Generator jest dostępny jako aplikacja okienkowa napisana w języku Java w środowisku IntelliJ IDEA Community.

Komisja Europejska definiuje dane osobowe jako „wszelkie informacje dotyczące zidentyfikowanej lub możliwej do zidentyfikowania żyjącej osoby fizycznej. Poszczególne informacje, które w połączeniu ze sobą mogą prowadzić do zidentyfikowania tożsamości danej osoby, także stanowią dane osobowe." Przykładowymi wymienianymi danymi osobowymi są imię i nazwisko, adres zamieszkania, adres e-mail taki jak imię.nazwisko@mail.com (ale już nie typu info@firma.com), dane o lokalizacji czy adres IP. Na podstawie definicji Komisji Europejskiej zdecydowałam, że dane generowane w moim projekcie będą uwzględniały:

- imię,
- nazwisko,
- płeć,
- datę urodzenia,
- kraj pochodzenia,
- adres zamieszkania (kraj, miasto, ulica, numer domu, kod pocztowy),
- numer telefonu,
- adres e-mail generowany na podstawie imienia i nazwiska.

Część danych jest generowana na podstawie innych, już wygenerowanych, danych, np. generowane imię pasuje do płci i kraju pochodzenia, a numer telefonu jest zgodny z zasadami tworzenia tego numeru w wygenerowanym kraju zamieszkania.

Wygenerowane dane można indywidualnie ponownie generować, bez zmieniania pozostałych informacji. Dane można zapisać na dysku w postaci pliku tekstowego.

2 Wymagania funkcjonalne

2.1 Historie użytkownika

1. Jako użytkownik chcę móc wygenerować sztuczne dane osobowe, aby mieć przykładowe dane osobowe.
2. Jako użytkownik chcę móc wygenerować spójne i logiczne dane, aby przypominały one jak najbardziej prawdziwe dane osobowe.

3. Jako użytkownik chcę móc zmienić poszczególne dane, aby mieć większy wpływ na wygenerowany wynik.

4. Jako użytkownik chcę móc zapisać wygenerowane dane, aby móc je wykorzystać w przyszłości.

2.2 Opis kluczowych funkcjonalności

Na podstawie historii użytkownika wyodrębniłam następujące funkcjonalności:

- Aplikacja generuje sztuczne dane osobowe zgodne z realnym światem, tzn. generowane kraje, adresy i imiona są prawdziwe, a numer telefonu jest generowany zgodnie z przepisami konkretnych państw.
- Użytkownik uruchamia generowanie danych poprzez naciśnięcie konkretnego przycisku.
- Naciśnięcie przycisku do generowania, gdy są już wygenerowane dane, powoduje wygenerowanie nowych danych i nadpisanie poprzednich.
- Użytkownik ma możliwość wygenerowania ponownie poszczególnych danych.
- Użytkownik jest proszony o potwierdzenie swojej decyzji dotyczącej ponownego wygenerowania danych, zarówno gdy ponownie będzie generował wszystkie, jak i niektóre pojedyncze dane.
- Użytkownik może zapisać wygenerowane dane na dysku.
- Wygenerowane imię pasuje do wygenerowanego kraju pochodzenia i płci.
- Wygenerowane nazwisko pasuje do wygenerowanego kraju pochodzenia.
- Bardziej prawdopodobne jest wygenerowanie imion i nazwisk często spotykanych w danym kraju.
- Wygenerowany numer telefonu pasuje do kraju zamieszkania.
- Wygenerowana data urodzenia jest maksymalnie lokalną datą w momencie generowania.
- Wygenerowana data urodzenia nie może mieć roku dalszego niż 200 lat wstecz od lokalnej daty w momencie generowania.
- Ponowne generowanie pojedynczych danych nie zmienia pozostałych danych, czyli np. ponownie wygenerowana płeć nie zmieni imienia.

3 Wymagania niefunkcjonalne

Wymagania zostały podzielone zgodnie z metodą FURPS. Wymagania funkcjonalne zostały opisane w poprzedniej sekcji, a pozostałe poniżej.

3.1 Użyteczność

- Interfejs aplikacji jest czytelny i przejrzysty.
- Generowane dane sa ułożone w dwie kolumny.
- Aplikacja jest utrzymana w stonowanej kolorystyce.
- Aplikacja jest w języku angielskim.

3.2 Niezawodność

- Aplikacja nie będzie się zawieszać.
- Aplikacja nie jest zależna od dostępu do internetu.

3.3 Wydajność

- Dane są generowane natychmiastowo.

3.4 Wsparcie

- Użytkownik jest proszony o potwierdzenie ponownego generowania danych, aby zapobiec niechcianej utracie wyników.
- W przypadku ponownego generowanie pojedynczych danych użytkownik jest ostrzegany, że potencjalnie nowy wynik może być niespójny z pozostałymi wynikami.

4 Technologie

Aplikacja jest napisana w języku Java w środowisku IntelliJ IDEA Community 2022.2.2. Interfejs graficzny aplikacji jest zaimplementowany przy użyciu Java Swing.

5 Implementacja

Większość informacji jest generowana na podstawie plików tekstowych zawierających potrzebne dane. Taka implementacja pozwala na stosunkowo łatwe rozszerzenie danych możliwych do wygenerowania.

5.1 Imię i nazwisko

Imiona i nazwiska możliwe do wygenerowania znajdują się w pliku *names.txt*. To, jak jest skonstruowany, jest przedstawione na Rys.1.

Każdy kraj posiada więcej niż 30 imion. Pierwsze 30 imion stanowią najczęstsze

```
[
        {
                "country":"country1",
                "male":["name1","name2",...],
                "female":["name1","name2",...],
                "surnames":["surname1","surname2",...]
        },
        {
                "country":"country2",
                "male":["name1","name2",...],
                "female":["name1","name2",...],
                "surnames":["surname1","surname2",...]
        }
]
```


Rysunek 1: Struktura pliku *names.txt*

imiona w danym kraju i są posortowane zgodnie z częstotliwością ich występowania. Częstość występowania konkretnych imion została sprawdzona na stronie *forbears.io*, która posiada dane z 2014 roku. Pozostałe imiona są ułożone losowe i zostały pobrane z repozytorium *uinames*. Generowanie imienia zgodnie z częstością jego występowania przebiega następująco: najpierw generowana jest losowa liczba z zakresu 0-100; jeśli jej wartość zawiera się w przedziale 0-40, to zwracane jest jedno z pierwszych 10 imion; jeśli w przedziale 41-70, to jedno z drugiej dziesiątki; jeśli w przedziale 71-90, to jedno z trzeciej dziesiątki; w przeciwnym przypadku losowane jest jedno z pozostałych imion.

Każdy kraj posiada więcej niż 20 nazwisk. Pierwsze 20 nazwisk to najczęstsze nazwiska w danym kraju. Tak jak w przypadku imion, częstość występowań danych nazwisk została sprawdzona na stronie *forbears.io*, a pozostałe nazwiska zostały pobrane z repozytorium *uinames*. Nazwiska nie są rozróżniane na żeńskie i męskie formy, więc może się zdarzyć kombinacja np. "Anna Kowalski". Generowanie nazwisk przebiega podobnie do generowania imion, lecz zamiast podziału na 4 kategorie, jest podział na 3. Prawdopodobieństwo wybrania kategorii z 10 najpopularniejszymi nazwiskami wynosi 40%, wybranie kategorii z kolejnymi 10 najpopularniejszymi wynosi 35%, a wybranie pozostałych ma prawdopodobieństwo 25%.

W pliku *names.txt* znajdują się jedynie imiona i nazwiska z krajów możliwych do wygenerowania. W przypadku dodania nowych państw, można rozbudować ten plik.

5.2 Kraj pochodzenia i zamieszkania

Wszystkie możliwe kraje do wygenerowania, zarówno jako kraje pochodzenia jak i kraje zamieszkania, znajdują się w pliku *countries.txt*. Dostępne kraje to: Wielka Brytania, Niemcy, Polska, Portugalia, Hiszpania, Stany Zjednoczone.

Program jest skonstruowany tak, że można go łatwo rozszerzać o kolejne państwa.

W przypadku generowania kraju zamieszkania, jest bardziej prawdopodbne, że zostanie wygenerowany kraj pochodzenia, niż jakikolwiek inny. Prawdopodbieństwo tego, że kraj zamieszkania jest taki sam jak pochodzenia jest ustawione na 60%.

5.3 Adres zamieszkania

Wszystkie możliwe do wygenerowania adresy zamieszkania znajdują się w pliku *adresses.txt*. To, jak jest skonstruowany, jest przedstawione na Rys.2.

```
[
        {
                "country":"country1",
                "address":["address1","address2",...]
        },
        {
                "country":"country2",
                "address":["address1","address2",...]
]
```

Rysunek 2: Struktura pliku *addresses.txt*

Kraj zamieszkania jest generowany oddzielnie, następnie na jego podstawie losowany jest jeden z zapisanych adresów. Każdy adres zawiera ulicę, numer domu, kod pocztowy i miasto. Nazwy miast są w języku kraju, a nie w angielskim, czyli jest np. Warszawa, a nie Warsaw. Kolejność zapisywania danych w adresie również odpowiada krajowi, np. w Wielkiej Brytanii kod pocztowy jest podawany za miastem, a w Polsce przed.

5.4 Płeć

Możliwe jest wygenerowanie płci męskiej lub żeńskiej, przy czym te opcje są zapisane w liście w programie. Można rozszerzyć tę listę o np. niebinarność, lecz aby program dalej poprawnie działał, trzeba by do pliku z imionami dodać kolejną kategorię, poza *male* i *female*.

5.5 E-mail

Adres e-mail jest generowany na podstawie imiona i nazwiska. Mając Imię Nazwisko, może być on wygenerowany w jeden z następujących sposobów:

- imię.nazwisko@mail.com,
- i.nazwisko@mail.com,

- imięnazwisko@mail.com,
- nazwisko.imię@mail.com.

5.6 Numer telefonu

Numer telefonu jest generowany na podstawie kraju zamieszkania. W pliku *coutries.txt* są zapisane numery kierunkowe danych krajów oraz z ilu cyfr składają się lokalne numery. Posiadając te informacje, program losuje odpowiednią liczbę cyfr i łączy je z numerem kierunkowym.

5.7 Data urodzenia

Data urodzenia jest zgodna z kalendarzem gregoriańskim. Generowana jest ona stopniowo.

Najpierw generowany jest rok urodzenia, przy czym nie może on przekraczać roku daty lokalnej użytkownika, ani roku 200 lat wstecz. Następnie generowany jest miesiąc, przy czym jeśli wcześniej został wygenerowany obecny rok, to nie może on przekraczać miesiąca daty lokalnej.

Jako ostatni generowany jest dzień urodzenia. Jeśli wcześniej został wygenerowany rok i miesiąc daty lokalnej, to dzień nie będzie przekraczał dnia daty lokalnej. Podczas generowania dnia jest też brane pod uwagę, to ile maksymalnie dni ma wygenerowany miesiąc, a jeśli wygenerowanym miesiącem był luty, to program sprawdza czy był to rok przestępny.

5.8 Zapisywanie

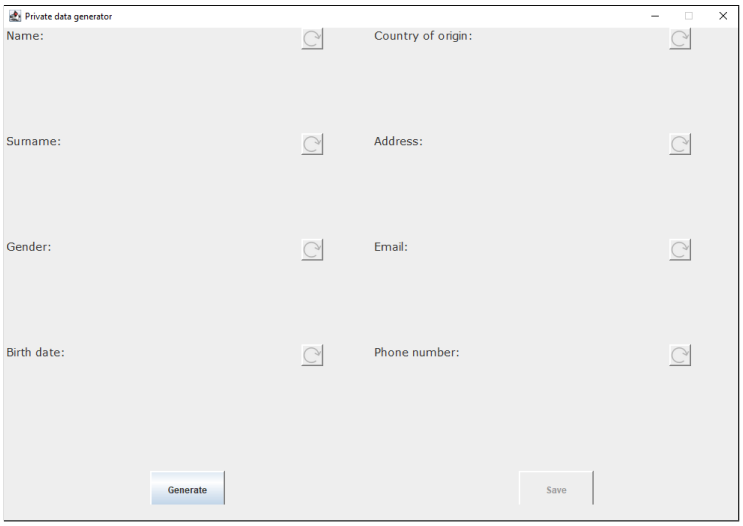
Użytkownik ma możliwość zapisania wygenerowanych danych na dysku. Po kliknięciu odpowiedniego przycisku otwiera się okno wyboru lokalizacji i nazwy pliku. Dane są zapisywane w postaci przedstawionej na Rys 3.

6 Interfejs użytkownika

Okno aplikacji ma stały rozmiar 1000 x 700 pikseli. Zawsze otwiera się na środku monitora użytkownika. Od razu po uruchomieniu nie ma żadnych wygenerowanych danych, a przyciski ponownego generowania i zapisywania są wyłączone, co można zobaczyć na Rys. 4.

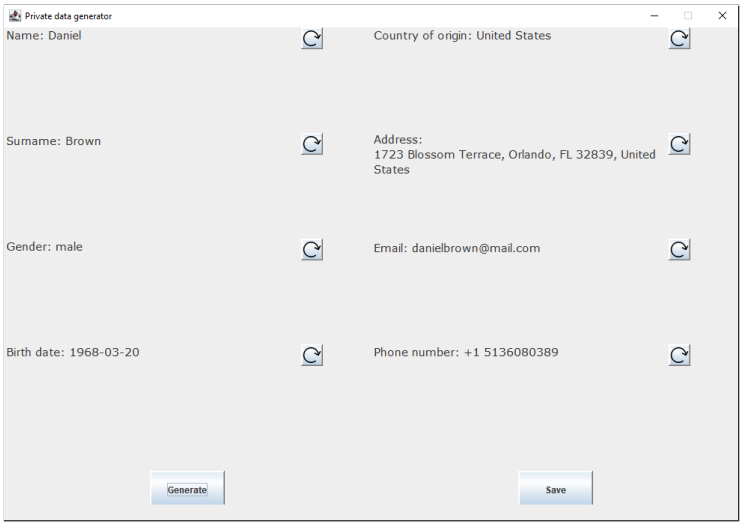
```
{
        "firstName":"name",
        "surname":"surname",
        "gender":"gender",
        "birthDate":"YYYY-MM-DD",
        "countryOfOrigin":"country",
        "address":"address",
        "email":"mail@mail.com",
        "phoneNumber":"phone number"
}
```

Rysunek 3: Struktura zapisanego pliku z danymi



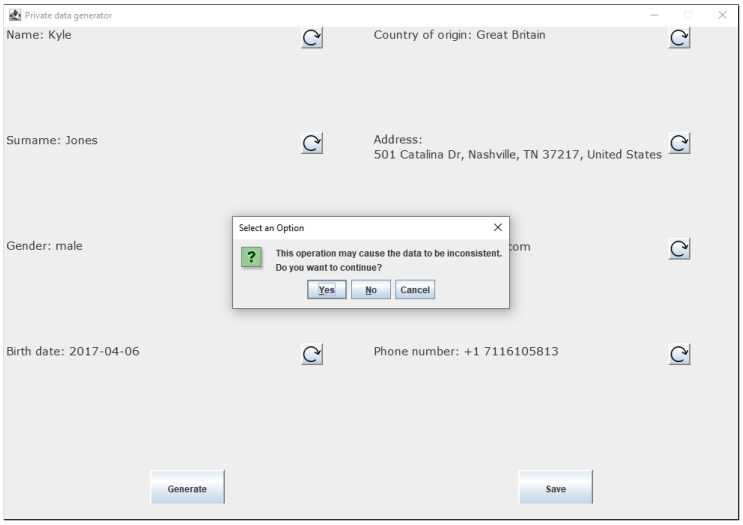
Rysunek 4: Wygląd aplikacji tuż po uruchomieniu

Po naciśnięciu przycisku **Generate** pojawiają się wygenerowane dane, co można zobaczyć na Rys. 5. Ikona użyta na przyciskach do ponownego generowania została pobrana ze strony flaticon.com i jest autorstwa użytkownika Creatype.



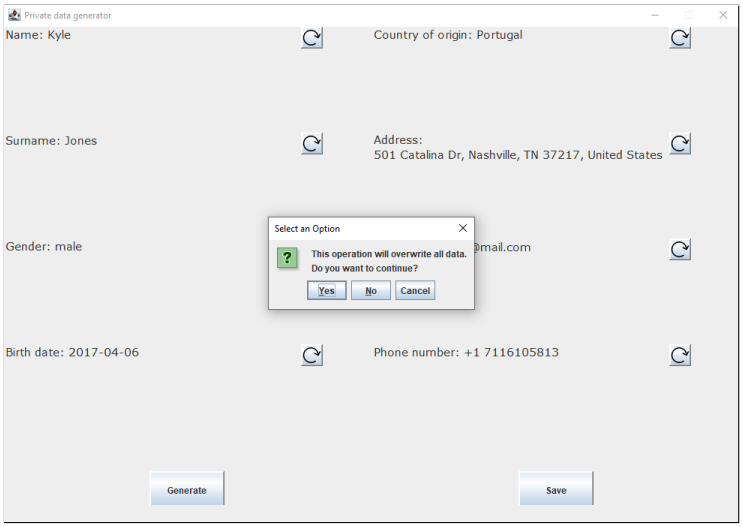
Rysunek 5: Wygląd aplikacji po wygenerowaniu danych

Jak użytkownik naciśnie przycisk do ponownego generowania pojedynczych danych, zmieniający informację, która potencjalnie może zepsuć jednolitość danych (płeć, kraj pochodzenia, adres), to wyświetla się okno proszące o potwierdzenie decyzji (Rys. 6).



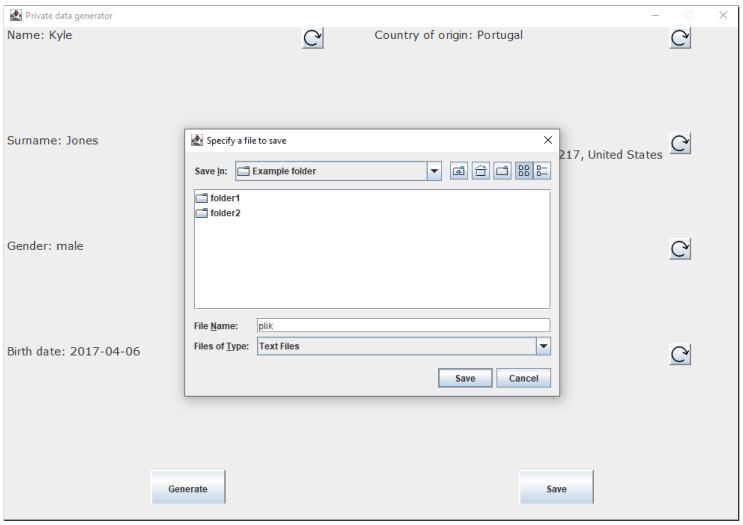
Rysunek 6: Pop-up proszący o potwierdzenie wygenerowania pojedynczej danej

Podobny pop-up pojawia się, gdy użytkownik ponownie kliknie przycisk **Generate**, jedynie zmienia się pierwsza linijka tesktu (Rys. 7).



Rysunek 7: Pop-up proszący o potwierdzenie wygenerowania wszystkich danych

Po kliknięciu przycisku **Save**, użytkownikowi pokaże się okno wyboru lokalizacji pliku do zapytania (Rys. 8). Użytkownik może zamknąć to okno i zapisywanie zostanie anulowane.



Rysunek 8: Okno zapisu

7 Źródła

1. Reforma unijnych przepisów o ochronie danych - Czym są dane osobowe?
2. Forebears: Names & Genealogical Resources
3. Repozytorium projektu *uinames.com*