



UDC 004.4

NEW TRANSLITERATION PROGRAM “COTOGWA” DEVELOPMENT

РОЗРОБКА НОВОЇ ПРОГРАМИ ТРАНСЛІТЕРАЦІЇ «COTOGWA»

Olena Surynovych¹, Natalia Lishchyna², Vladyslav Lohatskiy³, Maksym Savluk⁴, Solomiia Ryzhok⁵
Олена Суринович¹, Наталія Ліщин², Владислав Логацький³, Максим Савлук⁴, Соломія Рижок⁵

^{1,2,3,4,5} Lutsk National Technical University, Lutsk, Ukraine

ORCID: ¹<https://orcid.org/0000-0002-9300-0039>, ²<https://orcid.org/0000-0002-5200-536X>

E-mail: ¹sivom@ukr.net, ²lishchyna@gmail.com, ³v.lohatskiy@lntu.edu.ua,

⁴maswety8@gmail.com, ⁵s.ryzhok@lntu.edu.ua

Copyright © 2024 by author and the journal “Automation of technological and business – processes”.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0>



DOI: 10.15673/atbp.v16i3.2918

Abstract. The purpose of the work is to develop a program for transliteration and refactoring of mechanically typed text “CotoGwa”. A literary analysis of the research problem is presented. The choice of ways, technologies and means of solving the task is substantiated. The functional and structural scheme of the program is substantiated. The process of developing a software product is highlighted. Comparative characteristics of analogues are carried out and their shortcomings are described. The advantages of the developed product are defined too. The software development model and used scientific research methods are justified. The practical application of the created product is described step by step. Testing and debugging of the program is described.

Considering the implementation of the software, the architectural solutions and technologies used in the project were analyzed. The program takes into account the needs of users and can be easily extended to provide additional functionality thanks to the flexibility and extensive functionality of the C# programming language.

In the future, the development of the program will include additional options to create even greater convenience and speed of use by users. Taking into account feedback from users can contribute to the further development of this software in the field of text printing.

In general, the development makes it possible to efficiently and conveniently enter and edit text, reducing time spent on additional printing principles and facilitating routine tasks for users.

Анотація. Метою роботи є розробка програми для транслітерації та рефакторінгу механічно набраного тексту “CotoGwa”. Подано літературний аналіз проблеми дослідження. Обґрунтовано вибір шляхів, технологій і засобів вирішення поставленої задачі. Обґрунтовано функціонально-структурну схему програми. Висвітлено процес розробки програмного продукту. Проведено порівняльну характеристику аналогів та описано їх недоліки. Також визначено переваги розробленого продукту. Обґрунтовано модель розробки програмного забезпечення та використані методи наукового дослідження. Поетапно описано практичне застосування створеного продукту. Описано тестування та налагодження програми.

Розглядаючи реалізацію програмного забезпечення, було проаналізовано архітектурні рішення та технології, які використовуються в проекті. Програма враховує потреби користувачів і може бути легко розширена для забезпечення додаткової функціональності завдяки гнучкості та широкому функціоналу мови програмування C#.

Надалі в розробку програми будуть включені додаткові опції для створення ще більшої зручності та швидкості використання користувачами. Врахування відгуків користувачів може сприяти подальшому розвитку цього програмного забезпечення в області друку тексту.

Загалом розробка дозволяє ефективно та зручно вводити та редагувати текст, скорочуючи витрати часу на додаткові принципи друку та полегшуючи користувачам рутинні завдання.

Keywords: computer program, object-oriented programming (OOP), cross-platform, encapsulation, polymorphism, inheritance, programming, C#.

Ключові слова: комп'ютерна програма, об'єктно-орієнтоване програмування (ООП), кросплатформенність, інкапсуляція, поліморфізм, успадкування, програмування, C#.



1. Introduction

Various electronic devices have become an integral part of our daily life in the modern world of information technology. New challenges arise related to the speed and convenience of using technology. One such problem is incorrect typing on the keyboard, which can lead to wasted time and unexpected typing errors. In order to solve this problem, there is a need for a convenient and effective tool that provides users with the ability to quickly and easily correct text.

The product is created with the help of object-oriented programming language C#, which aims to facilitate the work of users who are closely related to writing a large amount of mechanical text, with help in its correction regardless of the keyboard layout. This program not only helps to correct mechanical errors when entering text, but also allows users to adjust various parameters, in particular, custom layouts.

One of the key features of the program is its flexibility and extensibility, as well as its cross-platform nature. The user can easily add and customize layouts to suit their unique needs, and use this program in any keyboard environment. This software includes many intelligent features, such as automatic text case conversion, changing themes and languages, the ability to manage symbols in the text, as well as optimizing the spaces in the entered text. This allows users to easily adapt the program to their needs and create text according to international standards.

An interesting addition to the product was the name transliteration function, which allows you to translate Ukrainian names and surnames into English in accordance with international standards and thus simplifies the task of writing unique names in international texts and documents.

The development process of this software will be revealed in detail, aimed at solving current problems in the field of word processing, and its potential for improving the efficiency and comfort of users in their everyday tasks will be shown too.

2. Literature analysis

One of the most widely used methods of conveying the form of a loan word is transliteration. Transliteration is the exact conveying of the letters of the word in the source language by means of the alphabet of the target language. This transformation may become a complicated process as the graphical systems of the source language and the target language are different. Erika Esenova [1], Rytsar B., Rozhankivskiy R., Mykulchik R. [2] dealt with the problem of transliteration. Not many other scholars researched the problem of the software developing in transliteration or automatic switching between different keyboard layouts. So, developed software can be a useful tool for increasing the convenience of working with a computer.

2.1. Description of analogues

Punto Switcher is a free program for automatically switching between different keyboard layouts in Windows and Mac OS X. The first version of the program, was published on September 14, 2001 (see fig. 1). The task of the program is to increase the convenience of working with a computer. In the background, it performs a statistical analysis of the sequences of characters entered, and if the combination of letters turns out to be atypical for the language in which the characters are entered, the program switches the language, erases what has been printed, emulating pressing the “Backspace” key, and re-enters the text already with the correct keyboard layout [3].

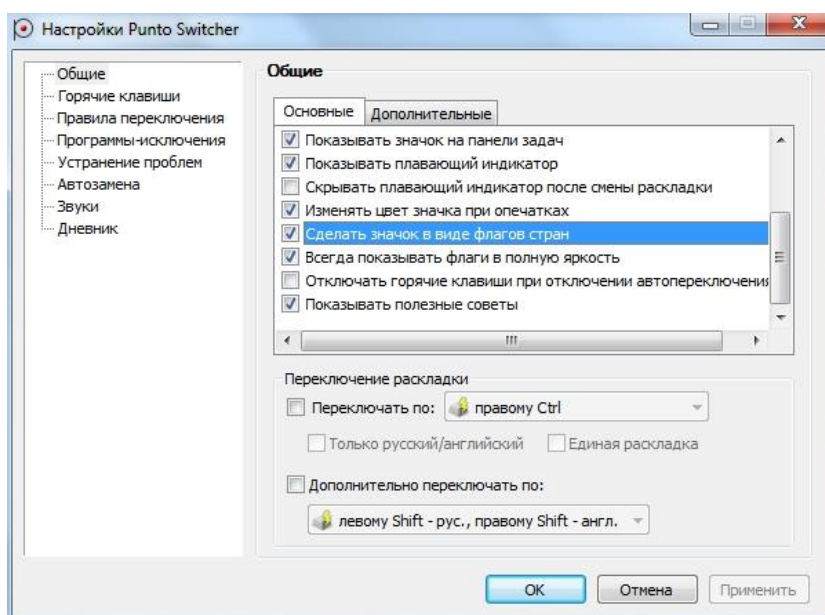


Fig. 1 – Punto Switcher Program

Another analogue is the program “Key Switcher” (see fig. 2). This is a program for the Windows operating system designed to automatically switch the keyboard layout during the typing. Main features of Key Switcher: Fixes common typos; fixes accidental double caps; recognizes suffixes in numbers; fixes the situation with the wrong case of letters; shows the corresponding country flag icon when changing the keyboard layout; smart undo to switch when the Backspace



key is pressed; quick call of an alternative layout; conversion of marked text; multilingual interface [4].

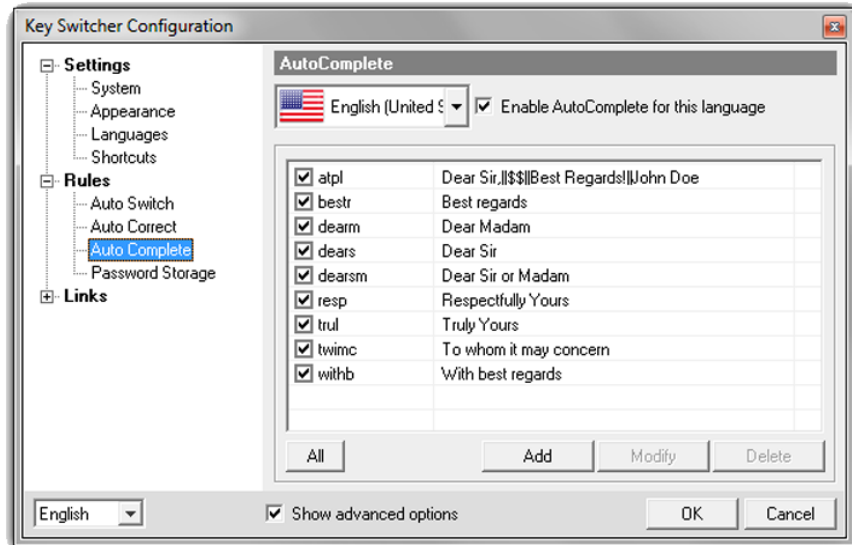


Fig. 2 – Key Switcher Program

Disadvantages of both programs:

- Punto Switcher is a product of a Russian company market. The economy of the aggressor country should not receive a single penny from the Ukrainian consumer;
- The program “Key Switcher” replaces the entered text, but at the same time requires manual prescription all substitution rules.

Advantages of our developed program:

- free;
- convenient interface;
- ability to specify independently key combination;
- the possibility of customizing your own layouts;
- the possibility of choosing a topic;
- ability to choose a language system interface.

3. Object, subject, and methods of research

The object of research is the software product (transliteration program “CotoGwa”) for different users who often use texts’ typing.

The subject of study is the process and tools of software development for working with text input and keyboard layout. “Spiral Model” methodology and survey methods werw used in the research.

The peculiarity of the approach (“Spiral Model”) is the emphasis on into four quadrants: identifying and understanding requirements, performing risk analysis, building the prototype and evaluation of the software's performance. (see fig. 3).

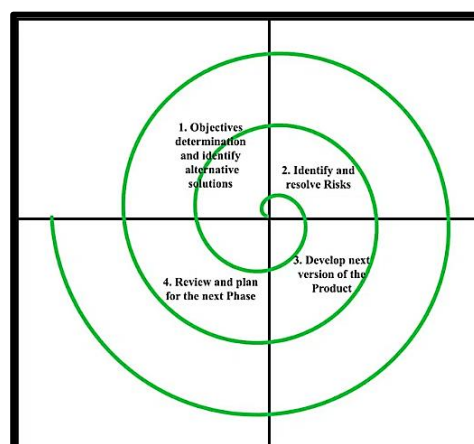


Fig. 3 – The Spiral Model [5]

The survey of 200 respondents, who can be potential users, was created. A poll was conducted among university students and teachers. 60 % of students and 40 % of teachers are interested of the development (see fig. 4).

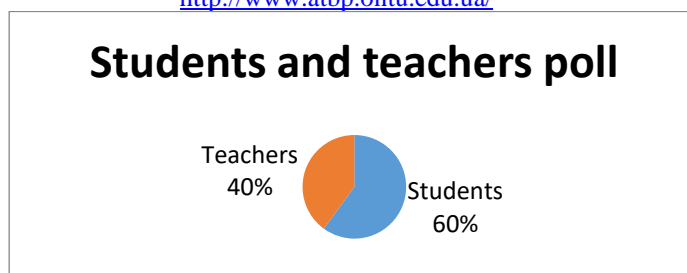


Fig. 4 – Students and teachers poll according to the interest in the software application

The survey of 50 respondents, who are faculty teachers, was created (see fig. 5). Among them are teachers who lead disciplines connected with English and teachers who often write English articles. 80 % of them are interested in our software development. 15 % of them want to use this application in their subjects immediately. 5% of them are not interested in our software development. So, this development can be usefull for their daily job.

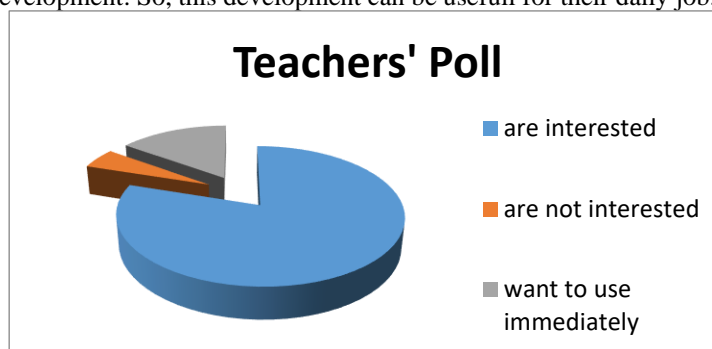


Fig. 5 – Teachers poll according to the interest in the software application

4. Results

4.1 Functional and structural diagram of the program

All features of the program are described below.

1) Autostart when the system is turned on. The first time the program is launched, the program turns on the “Autostart” switch, and now every time the system starts, the program will always be active in the background to execute and help with text correction. After turning on the “Autostart” switch, the program will not turn on and you will need to turn it on yourself for it to function.

2) Always active window in the tray. After the application is launched, it is active in the background, and when it is opened manually, it shows a static window. After minimizing or closing the window, the program will work and respond to commands given to it. To completely disable the program, you need to use the task manager or disable it yourself in the tray.

3) Database with settings. The program in the role of a database uses a JSON file, entering all program settings there (see fig. 6).

```

settings.json  [X]
Schema: <No Schema Selected>
1  {
2    "rusEgg": false,
3    "firstBoot": false,
4    "from": "en-US",
5    "to": "uk-UA",
6    "transHotkey": "Ctrl+Q",
7    "capsHotkey": "Ctrl+Oem6",
8    "isAutoStart": true,
9    "deleteSpaces": false,
10   "convertDouble": false,
11   "convertDoubleSpaces": false,
12   "defaultTheme": "Light"
13  }
    
```

Fig. 6 – Database with program settings

Next tasks are described here:

- selected folding transliteration (item 4-5);
- given key combinations (item 6-7);
- data of the first launch (item 3);



– data on the status of four functions of the main window (item 8-11).

Next functions are described too:

- “autostart”;
- “double character conversion”;
- “removing extra spaces”;
- “conversion of double spaces” and their status, where this feature is enabled for user use is shown.

4) Own key combinations to use the program are assigned.

In the main window, one of the 4 function keys is selected, such as: Ctrl, Win, Alt, Shift and some additional key.

Only up to 3 function keys can be combined, but do not forget about the additional auxiliary one. However, it should be noted that some key combinations are used in other programs, which may interfere with the operation of our product.

5) Transliteration. First, let's consider a small block diagram of the transliteration mechanism (see fig. 7). When the program is launched, the transliteration language from English to Ukrainian will be set automatically. It can be changed by clicking on the fields Base language (language from which transliteration) and Alternative language (language to be transliterated). Next, the key combination is selected, and the desired text is highlighted. After selecting the text, press the given key combination (1 or 2). The first combination is responsible for transliterating the text, and the second – for changing its register. After pressing the key combination, the program copies, processes and pastes already processed text. You can also turn on the function “Remove extra spaces”, which is related to these mechanisms. It will remove all spaces between words, leaving only one each.

6) “Remove extra spaces” function. This function removes extra spaces ($1 < x$) between words. This is a useful function for those, whose keyboard gets stuck, but it can be a hindrance to those, who have not learned how to make paragraphs in Word.

7) Function “Conversion of double spaces”. This function replaces a double space with a comma “,” as an example of how it works on smartphones. It is quite useful for people who quickly type text on a large scale.

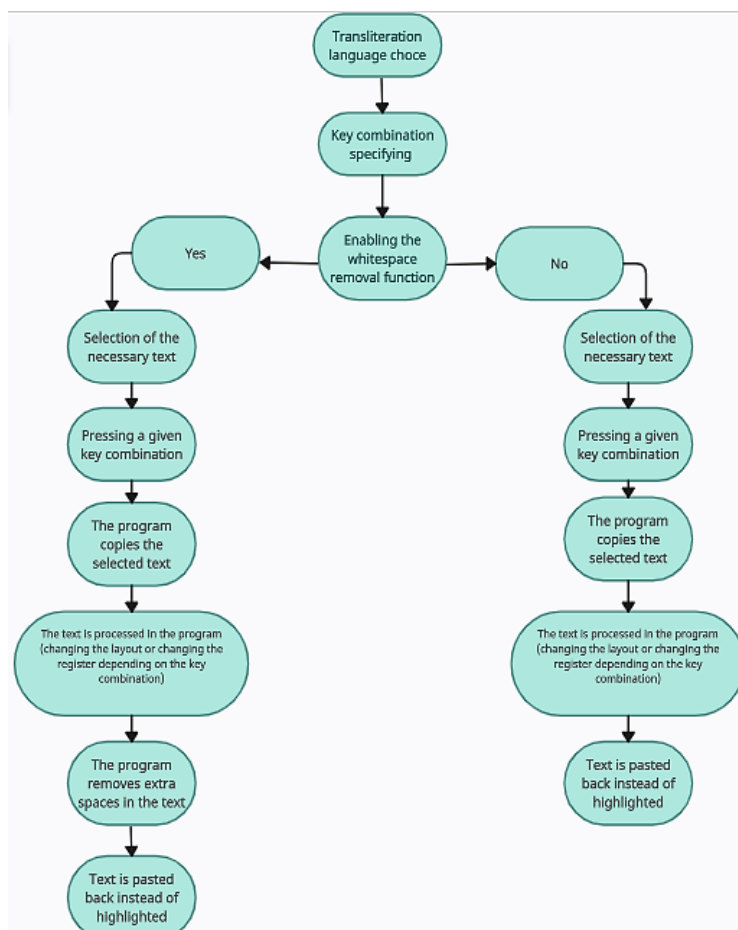


Fig. 7 – Block diagram of the transliteration mechanism

8) Function “Conversion of double characters”. This feature allows people to write characters from other layouts without changing the layout itself. For example, on key 3 we have two signs (№ on the Ukrainian layout) and (# on the English layout). A person, who writes text in English may need a symbol (№ on the Ukrainian layout) and instead of changing the layout, losing the focus of the cursor (in some text editors), she only needs to click twice on the desired symbol without changing the layout. For example, double-clicking ## will produce the character №.



<http://www.atbp.ontu.edu.ua/>

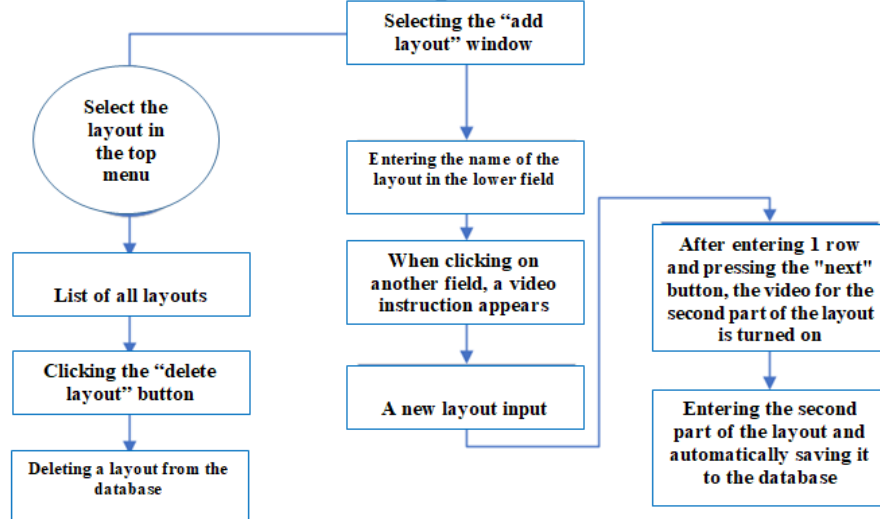


Fig. 8 – Layout editing scheme

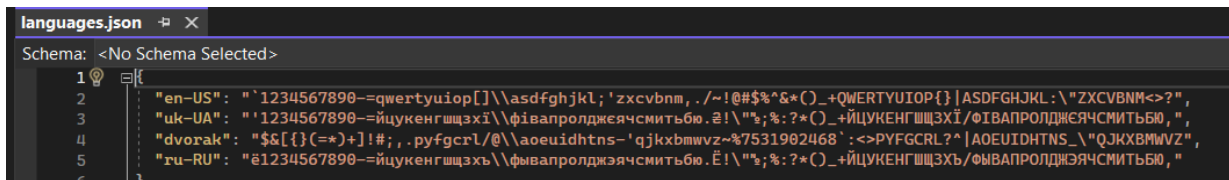


Fig. 9 – View of layouts in the database

10) Transliteration of names. This function works only in the middle of the program. The user enters his name and it is displayed in English according to international standards. This feature is useful for people who work with official documents.

11) Changing the theme and language of the program. The program allows you to choose one of two languages, English or Ukrainian. There is also a small selection of themes that change the visual color of this product.

4.2 Practical implementation of the program

Two fragments of the code of the main mechanisms of the program are described lower (see Listing 1-2).

Listing 1 – Fragment of transliteration code

```
public async Task<bool> tryTransliterateSelectedText()
{
    string textInClipboard = await taskManager.getTextFromClipboard();
    if (!(await taskManager.trySendKeys("⌘C"))) {
        return false;
    }
}
```

end of listing 1

This code shows how the transliteration mechanism works. First, this code receives the text, then it reads the parameters of the layout from which and to which it is transliterated, processes this text and inserts it in the place of what was highlighted.

Listing 2 – Fragment of reading settings code

```
internal class Source
{
    internal class Source
    {
        public static readonly string registryKey = "cotogwa_tess";

        public static JObject settings = Json.LoadJsonData(".||base\\settings.json");
        public static JObject lang = Json.LoadJsonData(".||base\\languages.json");
        public static JObject symbols = Json.LoadJsonData(".||base\\symbols.json");
        public static JObject easterEggs = Json.LoadJsonData(".||base\\easterEggs.json");
        public static JObject translit = Json.LoadJsonData(".||base\\translit.json");
        public static string[] jArrayKey = SetSettings.GetUIElementStrings(lang);
    }
}
```

end of listing 2



This fragment of the code shows ways to write and read program settings.

4.3 Program testing and debugging

Debug in Visual Studio was used to test and debug the program.

There was also manual testing. Manual testing of functions was done in different order, coming up with new combinations, tasks for the product.

Minimum PC requirements for using this program:

- Operating system: Windows 10/Windows 11;
- RAM: 512 mb;
- Processor: Intel Celeron – 3205u;
- Memory: 1GB.

4.4 Program usage

Examples of the program usage are described below.

The example of language changing, symbols is shown on the figures 10 and 11.



Fig. 10 – Written message

After pressing the keyboard shortcut for transliteration and case change with “remove extra spaces” enabled, the message looks like on the figure 11.



Fig. 11 - The message appearance after transliteration

The “Double Space Conversion” function is described below. Below is a message with a double space after the words (see fig. 12).



Fig. 12 – An example of replacing double spaces

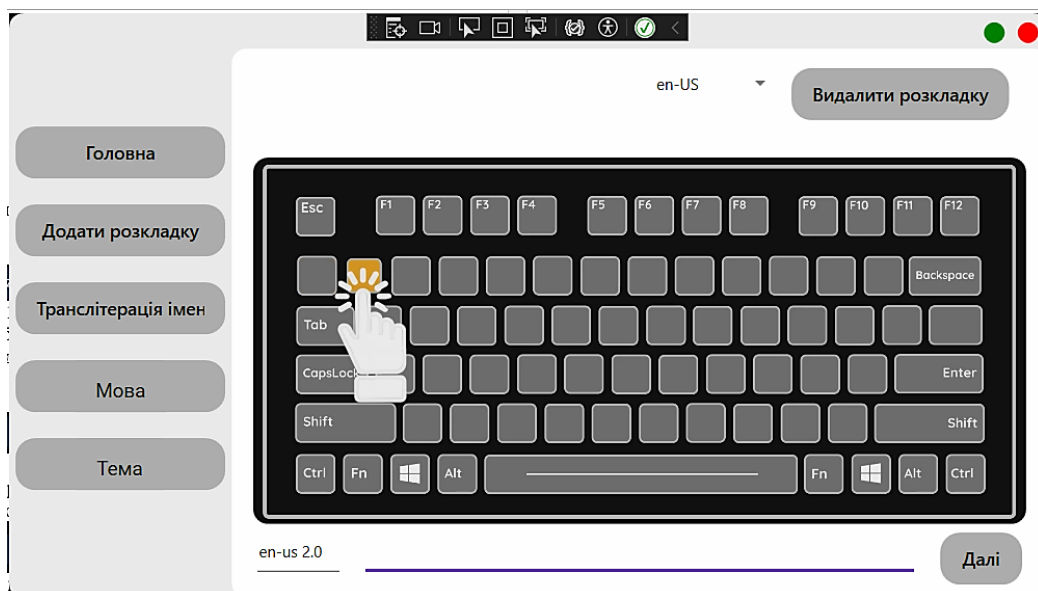


Fig. 13 – Window for adding layouts

The window for adding layouts is shown in the figure 13.

This window shows the ability to delete a layout, video instructions for adding a new layout, and the process of adding it.

The transliteration of the names is shown in figure 14.



<http://www.atbp.ontu.edu.ua/>

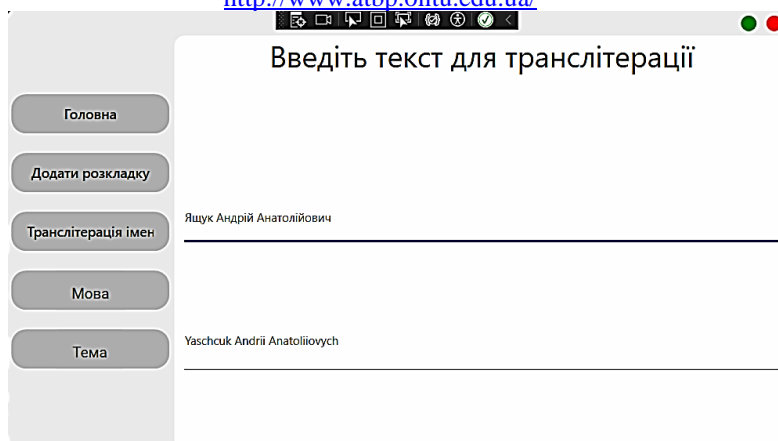


Fig. 14 – Example of names' transliteration

The final version of the product is shown in figure 15.

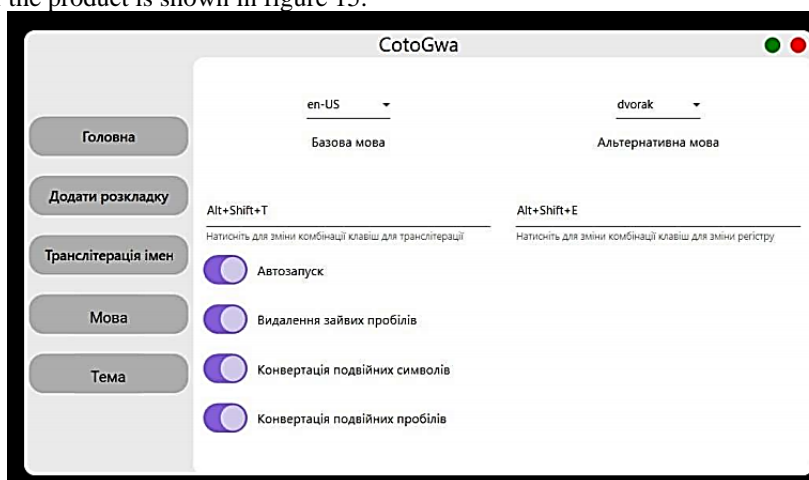


Fig. 15 – The final version of the product

5. Conclusions

The problem of incorrect mechanical text input was solved and software was developed in the C# programming language, aimed at improving the efficiency and convenience of text input. The developed program provides wide opportunities for adjusting the keyboard layout and provides a selection of functions for various needs.

One of the key advantages over the competition is the addition of custom layouts, name transliterations, and additional broad-based features. The program is not limited by the number of layouts, which allows users to customize this program according to individual needs. In addition, the function of transliteration of names from Ukrainian to English provides opportunities for convenient and accurate writing of initials in the international standard.

Considering the implementation of the software, the architectural solutions and technologies used in the project were analyzed. The program takes into account the needs of users and can be easily extended to provide additional functionality thanks to the flexibility and extensive functionality of the C# programming language.

In the future, the development of the program will include additional options to create even greater convenience and speed of use by users. Taking into account feedback from users can contribute to the further development of this software in the field of text printing.

In general, the development makes it possible to efficiently and conveniently enter and edit text, reducing time spent on additional printing principles and facilitating routine tasks for users.

6. References

- [1].Erika Esenova (2023). Transliteration as a means of conveying of the form of loan-words. *Transcarpathian Philological Studies*, 27 (2), 139-144. <https://doi.org/10.32782/tps2663-4880/2022.27.2.26>.
- [2].Rytsar B., Rozhankivskyi R., Mykulchyk R (2008). Transliteration of the Ukrainian texts from the Cyrillic into the Latin. *Technical Committee for Standardization of Scientific and Technical Terminology for State Consumer Standards of Ukraine and the Ministry of Education and Science of Ukraine*, 55-61. https://vlp.com.ua/files/10_38.pdf.
- [3].Punto Switcher (2024, January 15). Material from Wikipedia - the free encyclopedia. https://uk.wikipedia.org/wiki/Punto_Switcher.
- [4].Key Switcher 2.7 (2014, August 25). Material from Programy.com.ua. https://programy.com.ua/ua/key_switcher/.
- [5].Spiral Model – Software Engineering (2024, January 28). A computer science portal for geeks. <https://www.geeksforgeeks.org/software-engineering-spiral-model/>.

Отримана в редакції 26.07.2024. Прийнята до друку 17.08.2024. Received 26 July 2023. Approved 17 August 2024. Available in Internet 23 October 2024