

BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC KINH TẾ TP HỒ CHÍ MINH (UEH)
TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ



ĐỒ ÁN MÔN HỌC

ĐỀ TÀI

ỨNG DỤNG CẤU TRÚC DICTIONARY ĐỂ TẠO RA MỘT BỘ TỪ ĐIỂN ANH – VIỆT.

Học Phần: Cấu Trúc Dữ Liệu & Giải Thuật
Danh Sách Nhóm:

1. NGÔ GIA BẢO
2. PHẠM PHÚ KHÁNH
3. BÙI LÊ KHANG
4. TRƯƠNG THANH PHONG

Chuyên Ngành: **KHOA HỌC DỮ LIỆU**
Khóa: **K47**

Giảng Viên: TS. Đặng Ngọc Hoàng Thành

Tp. Hồ Chí Minh, Ngày 15 tháng 12 năm 2022

MỤC LỤC

MỤC LỤC	2
CHƯƠNG 1. DICTIONARY	4
1.1. Các Khái Niệm Liên Quan	4
1.2. Cấu Trúc và Cài Đặt Dictionary	4
1.2.1 Cấu trúc Dictionary	4
1.2.2 Cài đặt Dictionary	4
1.3. Các Thuật Toán Trên Dictionary	5
1.3.1 Thuật Toán Add	5
1.3.2 Thuật Toán Remove	5
1.3.3 Thuật Toán Duyệt	6
1.3.4 Thuật toán Contains	7
1.3.5 Thuật Toán Clear	7
1.3.6 Thuật Toán TryGetValue	7
CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ LỚP	9
2.1. Phân Tích Bài Toán Từ điển Anh-Việt	9
2.2. Sơ Đồ Lớp	10
2.3. Cài Đặt Lớp	15
CHƯƠNG 3. THIẾT KẾ GIAO DIỆN	17
3.1. Giao Diện Menu Chính	17
3.2. Chi Tiết Chức Năng	18
3.2.1. Từ yêu thích	18
3.2.2.Chọn loại từ điển.	18
3.2.2.a)Giao diện của từ điển Anh-Việt	19
3.2.2.b)Giao diện từ điển Việt-Anh.	20
3.2.3. Google dịch	20
3.2.4 Công cụ	21

3.2.4.a) 3000 từ Oxford	21
3.2.4.b) Dịch văn bản	22
3.2.5. Bàn phím ảo	23
3.2.6. Giới thiệu ứng dụng	23
CHƯƠNG 4. THẢO LUẬN & ĐÁNH GIÁ	24
4.1. Các Kết Quả Nhận Được	24
4.1.1. Thêm từ vào Từ Yêu thích bằng thuật toán Add.	24
4.1.2. Xóa từ vựng khỏi Từ Yêu Thích bằng thuật toán Remove.	26
4.1.3 Tra từ vựng	27
4.2. Một Số Tồn Tại	28
4.3. Hướng Phát Triển	28
PHỤ LỤC	30
TÀI LIỆU THAM KHẢO	31

CHƯƠNG 1. DICTIONARY

1.1. Các Khái Niệm Liên Quan

Khái niệm chung: Cấu trúc Dictionary được sử dụng trong nhiều loại ngôn ngữ khác nhau, bao gồm cả ngôn ngữ C#. Dictionary trong C# là một cấu trúc dữ liệu cho phép người dùng lưu trữ và truy cập dưới dạng cặp Key-Value. Với cấu trúc dạng cặp Dictionary thường lưu trữ các cấu trúc dữ liệu phức tạp như danh sách, cây, đồ thị để lưu trữ thông tin khách hàng, thông tin sản phẩm, Cấu trúc Dictionary cung cấp cho người một cách nhanh chóng và hiệu quả để tìm kiếm và truy xuất dữ liệu. Ngoài ra người dùng cũng có thể cập nhật dữ liệu trên Dictionary bất kể thời gian nào.

- **Key:** Mỗi một khóa sẽ xác định định một giá trị duy nhất và đặc biệt không được để trống.
- **Value:** Chỉ số của mảng dữ liệu được song ánh từ tập khóa, giá trị của Value có thể trùng lặp hoặc NULL.

1.2. Cấu Trúc và Cài Đặt Dictionary

1.2.1 Cấu trúc Dictionary

Dictionary là dữ liệu có cấu trúc theo dạng cặp Key-Value, trong đó với mỗi Key chỉ có một giá trị duy nhất và không được để trống. Điều này đồng nghĩa không thể xuất hiện 2 Key giống nhau trong một Dictionary. Value có giá trị có thể trùng lặp hoặc Null. Với liên kết theo dạng cặp người dùng có thể truy xuất giá trị của Value dựa vào Key liên kết với giá trị Value mà người dùng muốn truy xuất. Ngoài ra cấu trúc Dictionary còn cung cấp cho người dùng các phương thức như thêm, xóa và tìm kiếm.

1.2.2 Cài đặt Dictionary

using System;

using System.Collections.Generic;

public class Program

{

public static void Main()

{

```

        IDictionary<int, string> numberNames = new Dictionary<int,
string>();

    }

}

```

1.3. Các Thuật Toán Trên **Dictionary**

1.3.1 Thuật Toán Add

Ta dùng Thuật toán Add khi cần thêm dữ liệu (Key và Value) vào cơ sở dữ liệu.

Add(TKey Key, TValue Value)

- + Thuật toán Add được sử dụng để thêm một Key và Value đã chỉ định vào Dictionary.

CODE:

```

public static void Main()
{
    Dictionary<string, string> myDict =
        new Dictionary<string, string>();

    myDict.Add("Australia", "Canberra");
    myDict.Add("Belgium", "Brussels");
    myDict.Add("Netherlands", "Amsterdam");
}

```

1.3.2 Thuật Toán Remove

Mỗi phần tử đều có Key khác nhau, đại diện cho một Value (Value có thể trùng nhau). Do đó thuật toán được sử dụng để xóa Key và Value tương ứng (Key phải tồn tại trong cơ sở dữ liệu).

Remove(TKey Key)

- + Thuật toán Remove được sử dụng để xóa đối tượng có Key được chỉ định trong Dictionary

CODE:

```

public static void Main()

```

```

{
    Dictionary<string, string> myDict =
        new Dictionary<string, string>();

    myDict.Remove("Russia")
}

```

1.3.3 Thuật Toán Duyệt

Mỗi phần tử trong Dictionary (bao gồm một cặp Key - Value) được định nghĩa là một đối tượng. Vì vậy, ta có thể dùng **foreach** để duyệt 1 Dictionary và in ra giá trị Key-Value.

KeyValuePair<TKey, TValue>

- + **Key:** trả về giá trị Key hiện tại của phần tử.
- + **Value:** trả về giá trị Value hiện tại của phần tử.

CODE:

```

public static void Main()
{
    Dictionary<string, string> myDict =
        new Dictionary<string, string>();

    myDict.Add("Australia", "Canberra");
    myDict.Add("Belgium", "Brussels");
    myDict.Add("Netherlands", "Amsterdam");
    foreach (KeyValuePair<string, string> item in myDict)
    {
        Console.WriteLine(item.Key + "\t" + item.Value)
    }
}

```

1.3.4 Thuật toán Contains

Ta sử dụng thuật toán Contains để kiểm tra xem đối tượng đã tồn tại trong Dictionary hay chưa. Thuật toán Contains được dùng cho cả Key và Value, nếu có tồn tại thì kết quả trả về là True, ngược lại kết quả sẽ làm False.

ContainsKey(**TKey** Key)

ContainsValue(**TValue** Value)

CODE:

```
bool TH1 = MyDict.ContainsKey(1);
```

```
bool TH2 = MyDict.ContainsValue(Four);
```

```
Console.WriteLine("Ket Qua 1: " + TH1);
```

```
Console.WriteLine("Ket Qua 2:" + TH2);
```

1.3.5 Thuật Toán Clear

Khi muốn xóa toàn bộ dữ liệu trong Dictionary, ta dùng thuật toán Clear. Ví dụ, số phần tử trong Dictionary trước và sau khi xóa

Clear()

CODE:

```
Console.WriteLine("BanDau " + MyDict.Count());
```

```
MyDict.Clear();
```

```
Console.WriteLine("SauKhiXoa " + MyDict.Count());
```

1.3.6 Thuật Toán TryGetValue

Thuật toán giúp kiểm tra Key có tồn tại không. Nếu có trả về True đồng thời trả về giá trị Value tương ứng qua biến Value. Ngược lại trả về False. Từ thuật toán này và Contains, ta có thể xây dựng các phương thức tìm kiếm trong Dictionary.

TryGetValue(**TKey** Key, **TValue** Value)

CODE:

```
if (MyDict.TryGetValue("1", out value))
```

```
{
```

```
    Console.WriteLine("For key = \"1\", value = {0}.", value);  
}  
else  
{  
    Console.WriteLine("Key = \"1\" is not found.");  
}
```


CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ LỚP

2.1. Phân Tích Bài Toán **Từ điển Anh-Việt**

Đề: Ứng dụng cấu trúc Dictionary để tạo ra một bộ từ điển Anh-Việt.

INPUT: Nhập dữ liệu về từ vựng tiếng anh , tiếng việt.

OUTPUT: Xuất ra nghĩa tiếng anh và tiếng việt, phiên âm của từ vựng, loại từ.

Bước 1: Xây dựng giao diện winform cho từ điển .

Bước 2: Xây dựng giao diện winform các chức năng chuyển Anh-Việt thành Việt-Anh và ngược lại, Từ Yêu Thích, 3000 Oxford, Dịch văn bản, Google dịch, Bàn phím ảo của từ điển.

Bước 3: Tạo liên kết các chức năng với từ điển

Bước 4: Xây dựng file text cơ sở dữ liệu về từ điển Anh-Việt, Việt-Anh.

Bước 5: Xây dựng file text theo cấu trúc dữ liệu Dictionary dùng để đọc cơ sở dữ liệu ở trong tệp tin ở B1 theo cấu trúc <Từ vựng:Dòng bắt đầu đọc dữ liệu, Số dòng đọc dữ liệu> với Từ vựng theo kiểu dữ liệu string và (Dòng bắt đầu đọc dữ liệu, Số dòng đọc dữ liệu) có kiểu dữ liệu string.

Một phần tử trong Dictionary sẽ có cấu trúc <Key:Value> với kiểu dữ liệu được nhập trong file text ở bước 2 thì Từ Vựng sẽ là Key và (Dòng bắt đầu đọc dữ liệu, Số dòng đọc dữ liệu) là Value.

Bước 6: Xây dựng lớp Dictionary

```
class Dictionary
{
    public Dictionary<String, String> dictIndex = new Dictionary<string, string>();(Từ điển Anh-Việt)

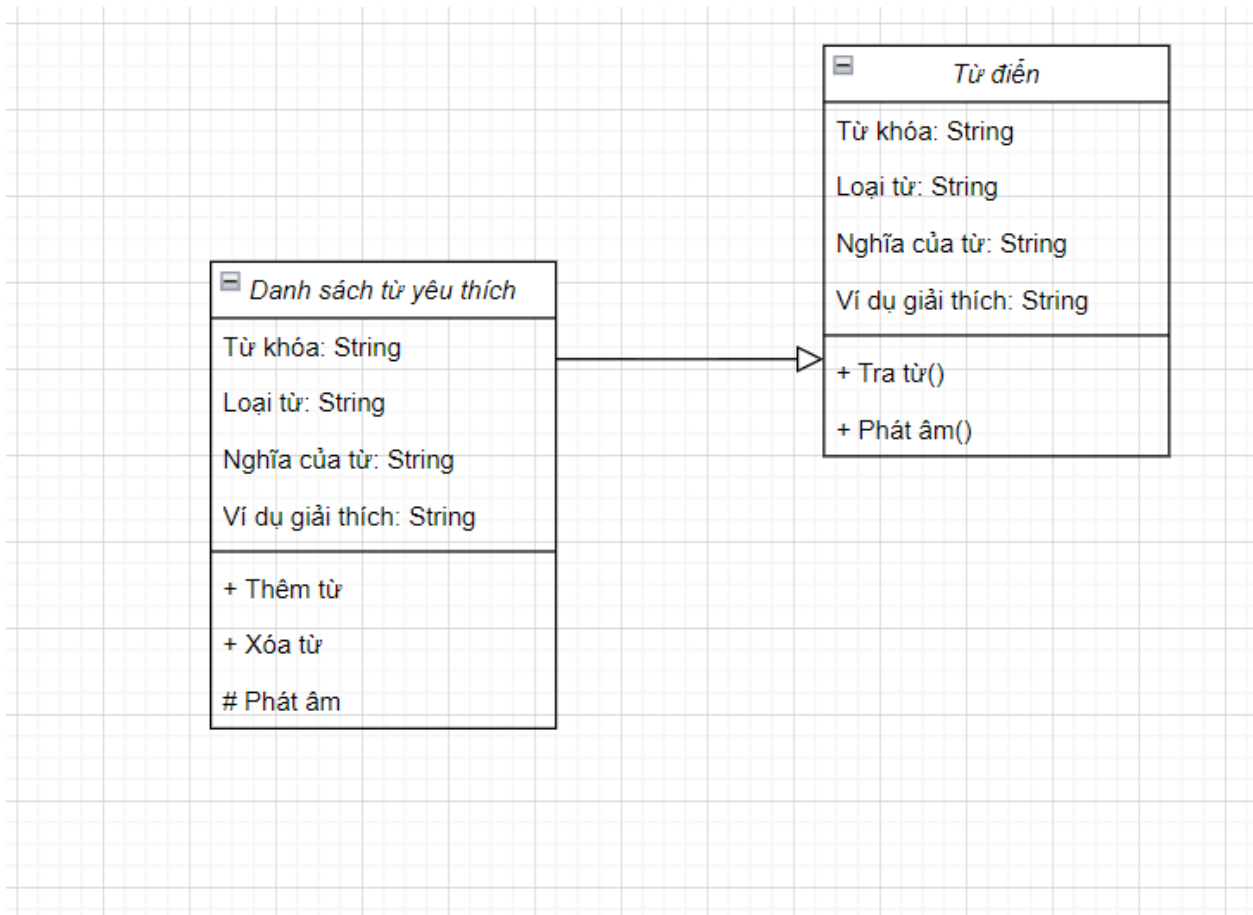
    public Dictionary<String, String> dictIndexVA = new Dictionary<string, string>();(Từ điển Việt-Anh)
}
```

Bước 7: Xây dựng các thuật toán của cấu trúc Dictionary trong từ điển Anh-Việt, Việt-Anh

- Thuật toán Add để thêm từ vào cơ sở dữ liệu Từ Yêu Thích.
- Thuật toán Remove để xóa từ ở cơ sở dữ liệu Từ Yêu Thích.
- Thuật toán Search để tra từ vựng trong từ điển.

2.2. Sơ Đồ Lớp

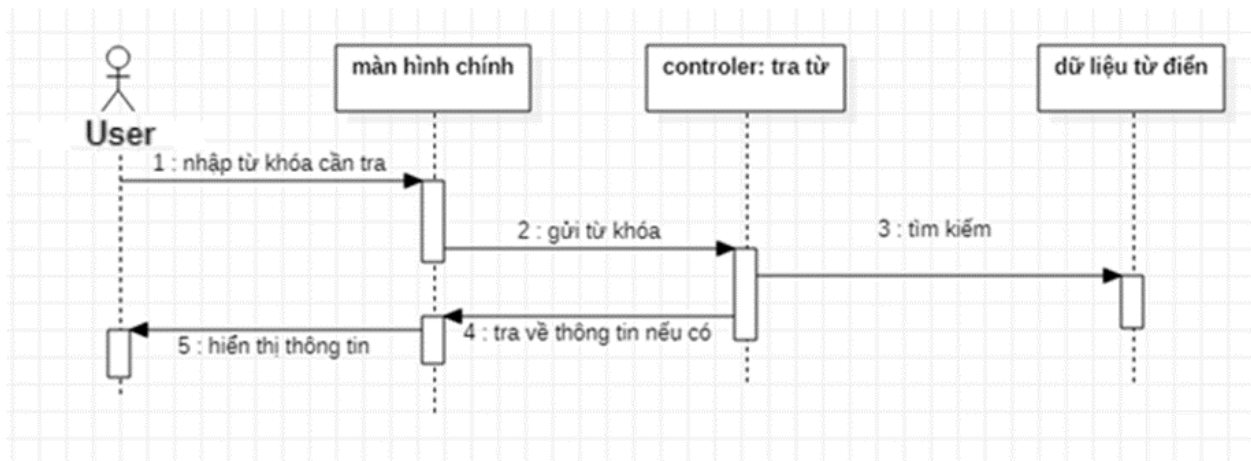
2.2.1 Biểu đồ lớp



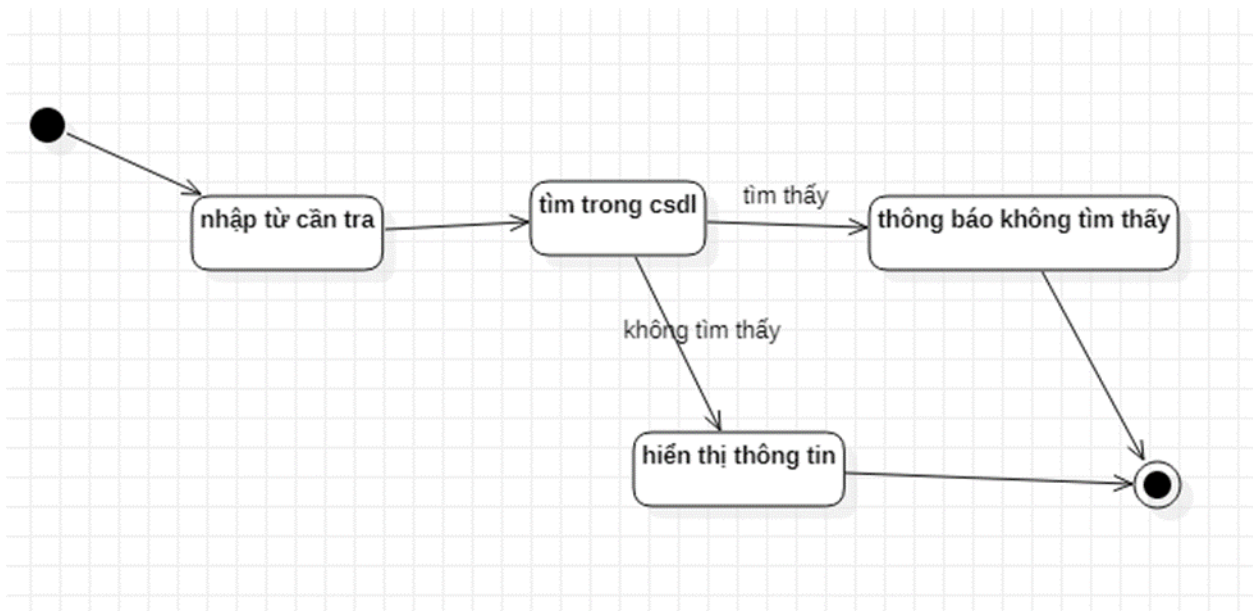
2.2.2 Biểu đồ hoạt động và biểu đồ trình tự:

Chức năng tra từ:

- Mục đích: người dùng truy cập vào phần mềm để tra cứu nghĩa của từ và các thông tin của từ, từ tiếng anh sang tiếng việt hoặc ngược lại.
- Mô tả: người dùng nhập từ cần tra vào ô tìm kiếm sau đó nhấn enter hoặc click vào các từ đã gợi sẵn trong listbox.



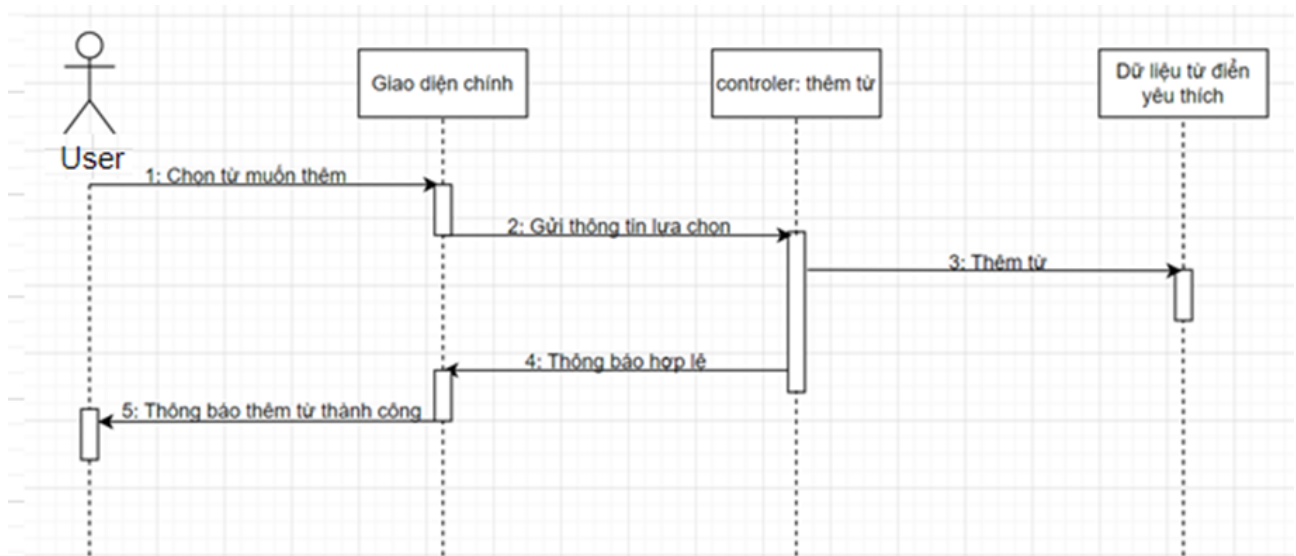
Biểu đồ trình tự cho chức năng tra từ.



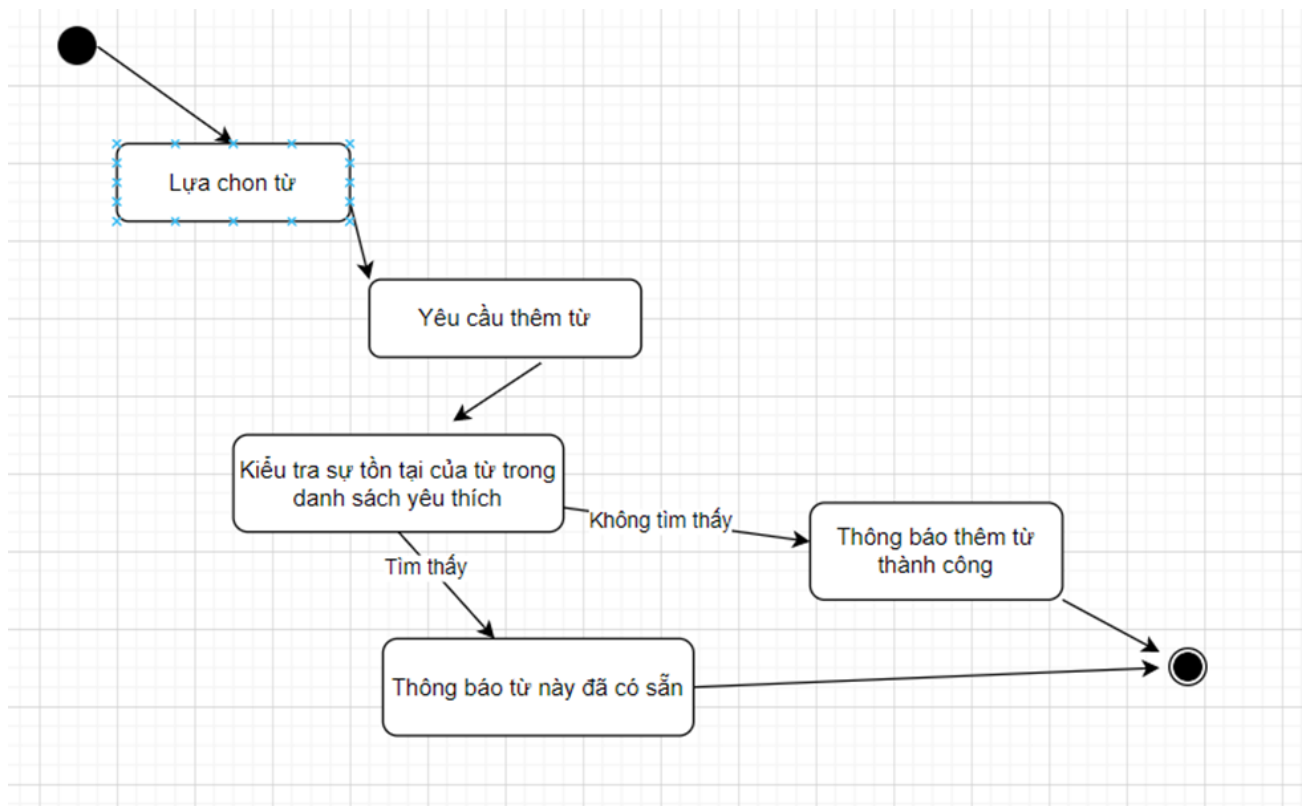
Biểu đồ hoạt động cho chức năng tra từ.

Chức năng thêm từ:

- Mục đích: Người dùng thêm dữ liệu từ vào cơ sở dữ liệu những từ ưa thích.
- Mô tả: Trên giao diện người dùng, người dùng có thể bấm chuột phải vào từ muốn thêm vào danh sách và chọn chức năng thêm vào danh sách yêu thích. Nếu từ này đã có trong danh sách ưu thích ngay từ trước thì sẽ thông báo lên thanh trạng thái từ này đã tồn tại trong danh sách nếu không thì thông báo thêm từ thành công.



Biểu đồ trình tự cho chức năng thêm từ ưa thích.

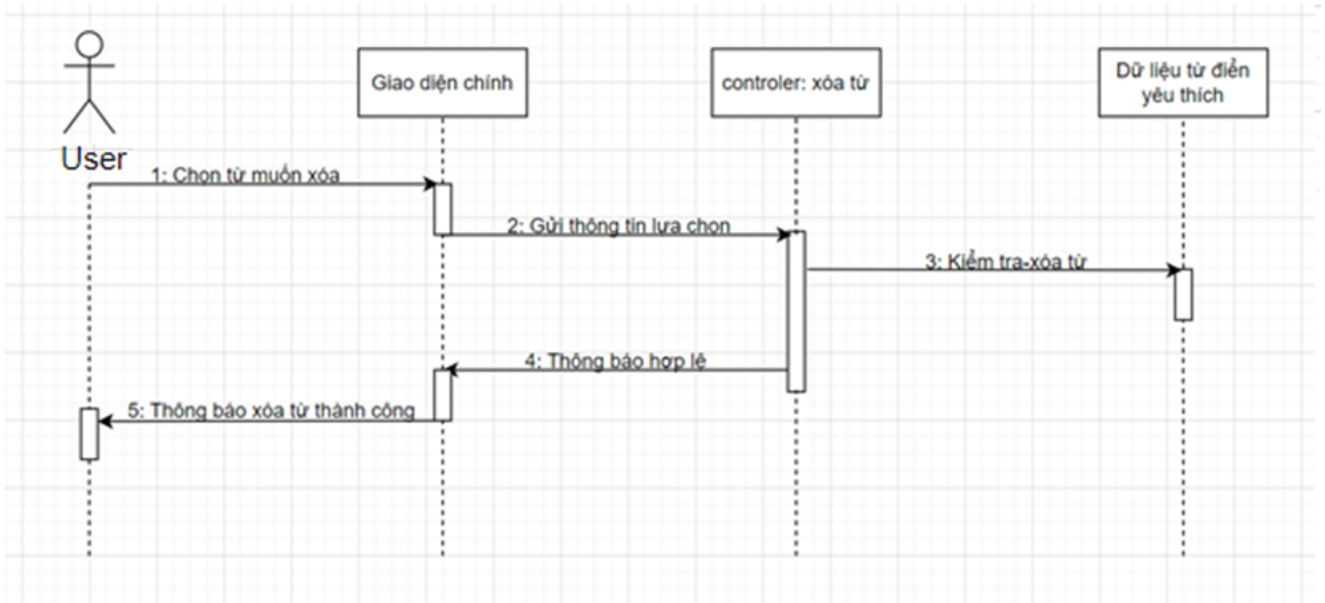


Biểu đồ hoạt động cho chức năng thêm từ ưa thích

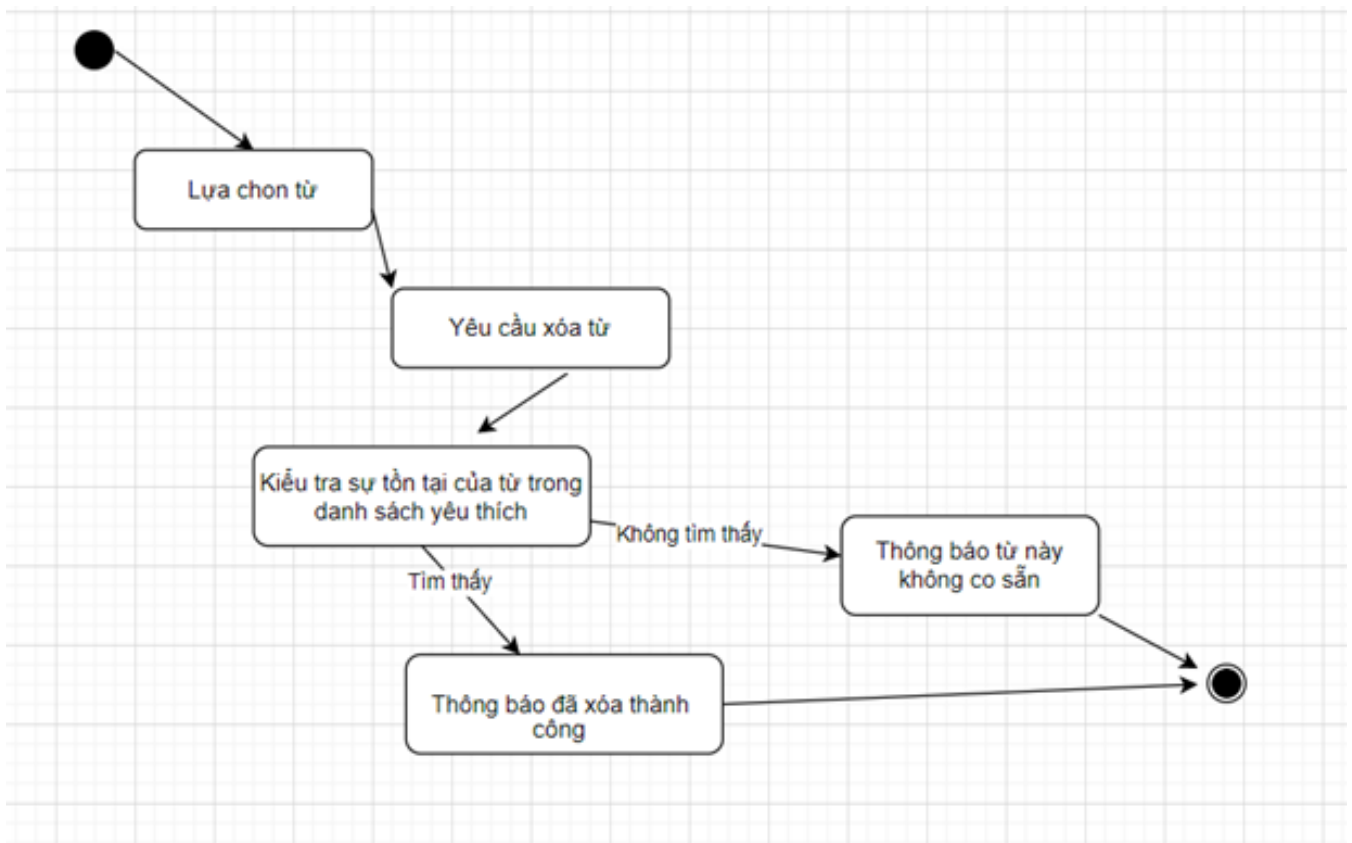
Chức năng xóa từ:

- Mục đích: Người dùng xóa dữ liệu từ trong cơ sở dữ liệu những từ ưa thích.

- Mô tả: Trên giao diện người dùng, người dùng có thể chọn một hoặc nhiều từ sau đó chuột phải và chọn remove để xóa khỏi cơ sở dữ liệu từ. Nếu xóa thành công thì sẽ thông báo lên thanh status.



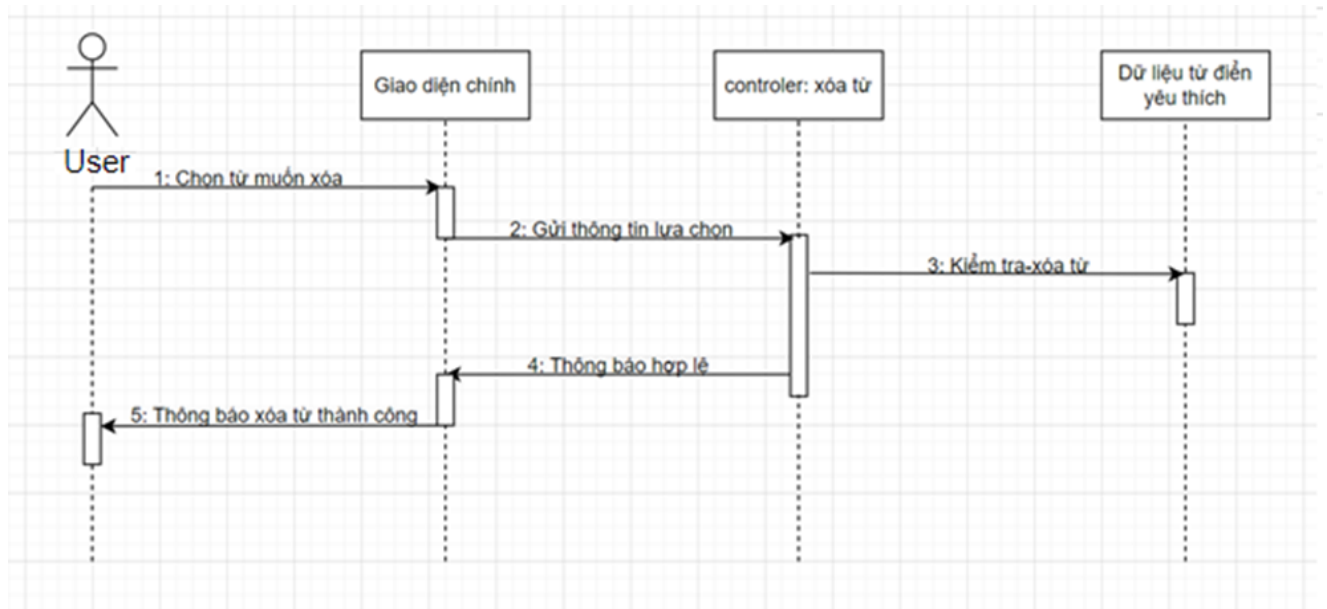
Biểu đồ trình tự cho chức năng xóa từ ưa thích



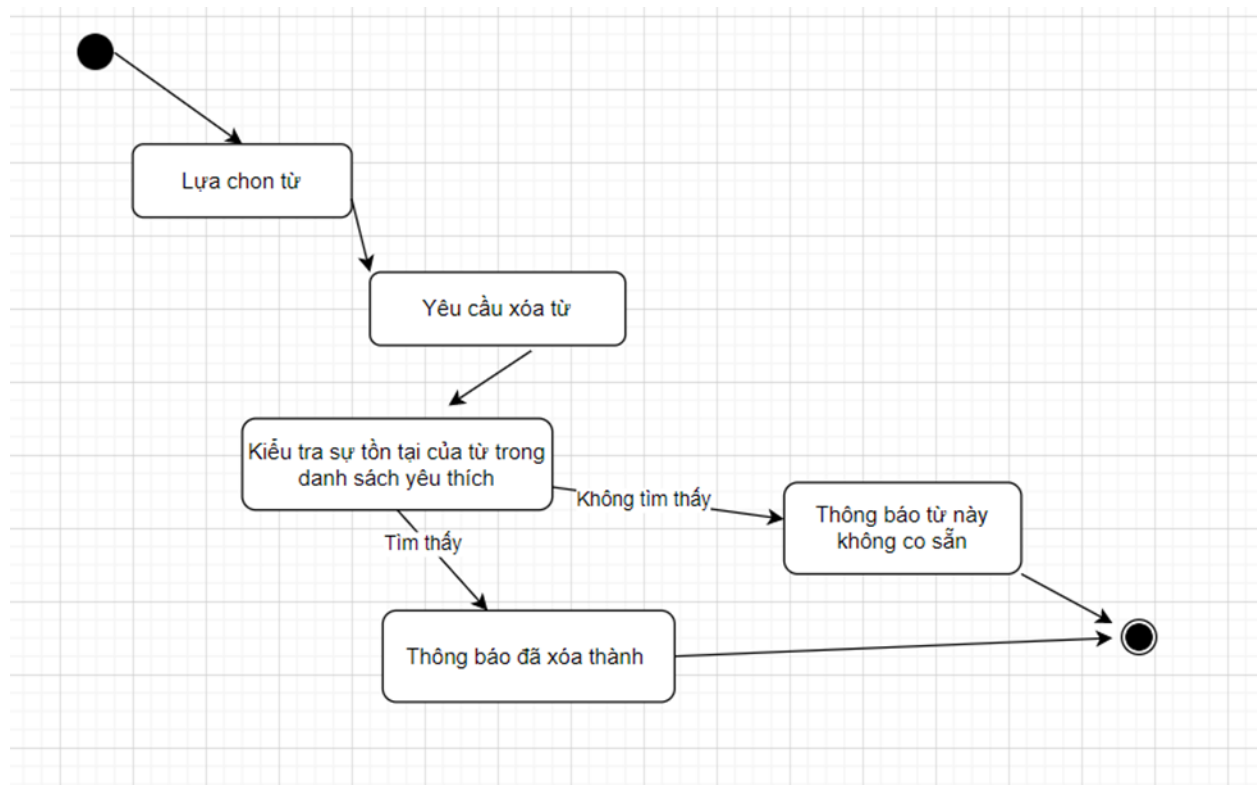
Biểu đồ hoạt động cho chức năng xóa từ ưa thích

Chức năng phát âm câu tiếng anh.

- Mục đích: phát âm ra câu tiếng anh hoặc từ tiếng anh mà người sử dụng muốn nghe.
- Mô tả: người sử dụng nhập từ tiếng anh muốn nghe phát âm, rồi click nút hình cái tai sau đó thì lắng nghe.



Biểu đồ trình tự cho chức năng phát âm câu tiếng Anh



Biểu đồ hoạt động cho chức năng phát âm câu tiếng Anh

2.3. Cài Đặt Lớp

2.3.1 Dữ liệu từ điển.

Dữ liệu từ điển được lưu trong 2 file text bao gồm 2 file ghi dữ liệu từ tiếng Anh và 2 file ghi dữ liệu tiếng Việt (1 file chứa từ, 1 file chứa ngữ nghĩa, loại từ, cách sử dụng...). dữ liệu được chia ra theo chữ cái đầu a-z. Và đối với file chứa các từ tiếng Anh, Việt sẽ bao gồm từ vựng và một chuỗi có ý nghĩa là bắt đầu đọc từ dòng này, ngăn cách bởi dấu “:” số dòng đọc dữ liệu kể từ vị trí bắt đầu. ví dụ: abandonment:165,4. Còn đối với file chứa dữ liệu về nghĩa của từ, loại từ,... thì luôn bắt đầu với @ với mỗi từ. Sau đó là toàn bộ thông tin về từ đó.

2.3.2 Mô tả quá trình đọc dữ liệu từ file ra:

Để đọc và ghi file ta sử dụng lớp Filestream của namespace system.io trong C# hỗ trợ việc đọc và ghi file.

Đầu tiên chúng ta sẽ tạo một đối tượng Filesteam truy xuất đến file và một StreamReader (reader) để đọc file. Khởi tạo vòng lặp với điều kiện reader.Peek() ≥ 0 và khai báo biến string s, cho đọc từng dòng dữ liệu vào và đưa dữ liệu vào s, dùng phương thức split để tách thành các trường dữ liệu sau đó tiến hành add vào list và dictionary được khai báo từ trước.

2.3.3 Mô tả giải thuật tra từ và nghĩa của từ:

Khi người sử dụng nhập từ cần tra, phần mềm sẽ dựa vào dữ liệu được nhập vào và tìm trong danh sách từ. Sau đó xác định chỉ số dòng bắt đầu chứa dữ liệu liên quan đến từ và số dòng chứa dữ liệu. Tiến hành cho xuất dữ liệu lên màn hình.

2.3.4 Mô tả giải thuật thêm từ vào danh sách từ yêu thích.

Dựa vào từ khóa người sử dụng nhập phần mềm sẽ tìm ra thông tin của từ sau khi lựa chọn thêm từ vào danh sách ưa thích thì chương trình sẽ kiểm tra sự tồn tại của từ trong database của danh sách yêu thích. Nếu chưa tồn tại thì thêm vào, nếu không thích thông báo từ này đã có trong danh sách yêu thích.

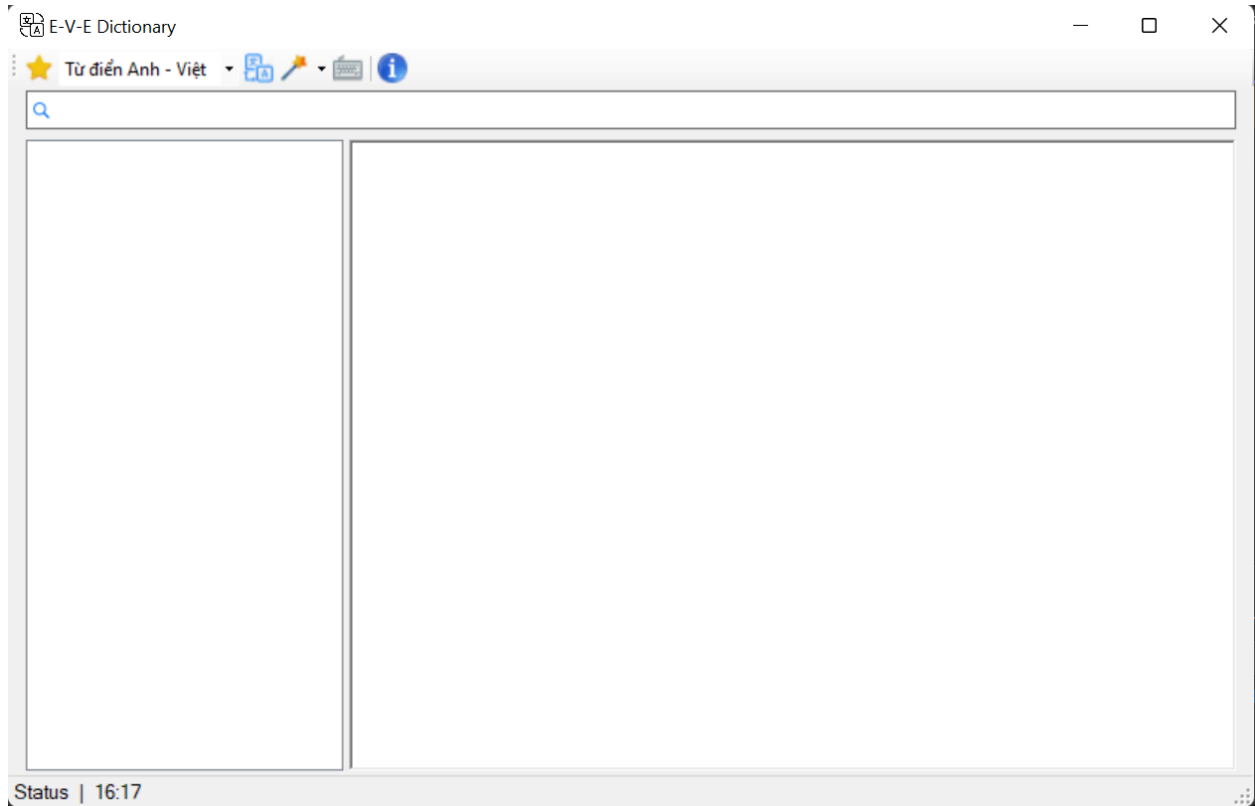
Mô tả giải thuật xóa từ trong danh sách yêu thích

Phần mềm sẽ dựa vào từ người sử dụng lựa chọn để tiến hành xóa khỏi database của từ yêu thích.

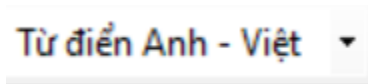
Note: Code cài đặt đã được đăng lên github

CHƯƠNG 3. THIẾT KẾ GIAO DIỆN

3.1. Giao Diện Menu Chính



: Từ Yêu Thích.



: Loại từ điển.



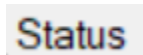
: Google dịch.



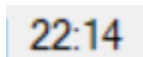
: Công cụ.



: Bàn phím ảo.



: Trạng thái của từ điển.



: Giờ.



: Giới thiệu ứng dụng

3.2. Chi Tiết Chức Năng

3.2.1. Từ yêu thích

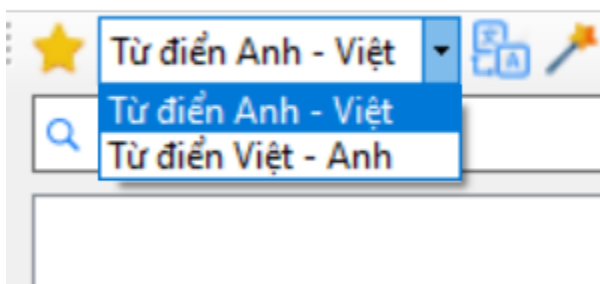


a-going : status thể hiện từ đang dịch

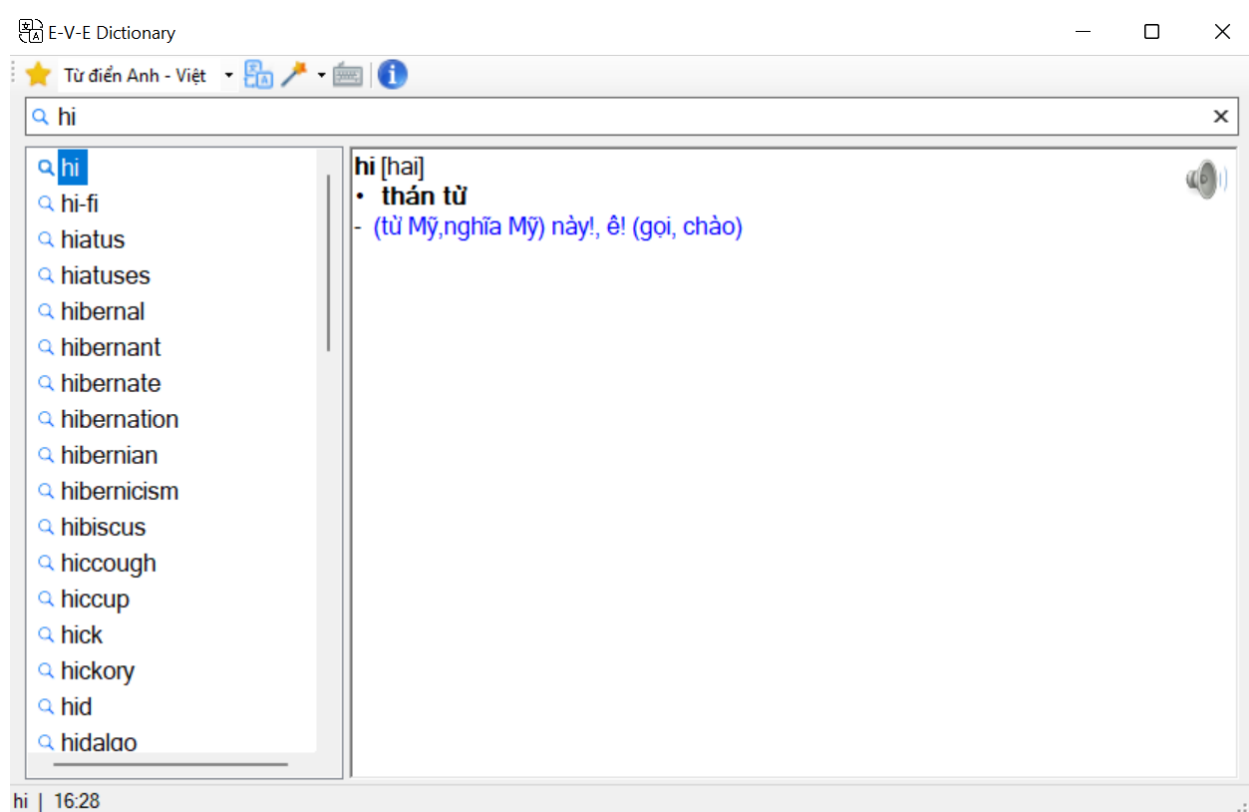
3.2.2. Chọn loại từ điển.



E-V-E Dictionary

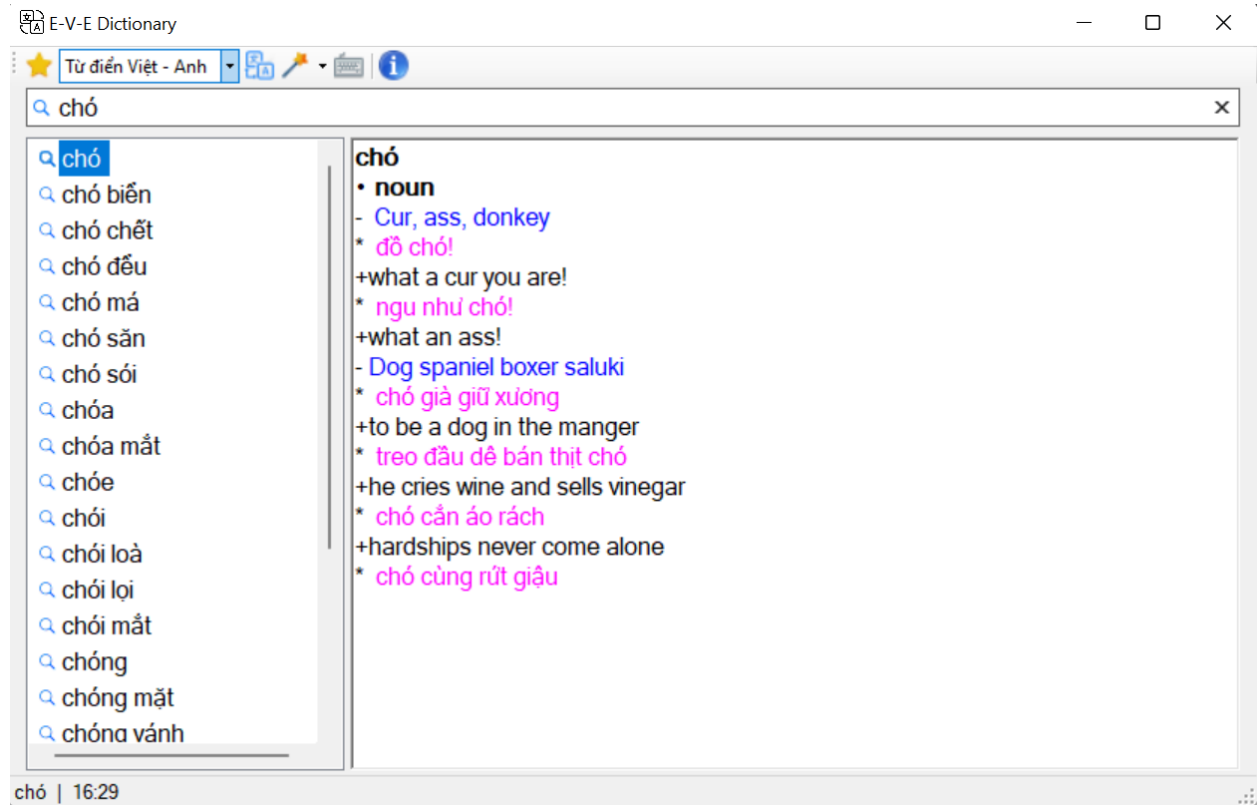


3.2.2.a) Giao diện của từ điển Anh-Việt



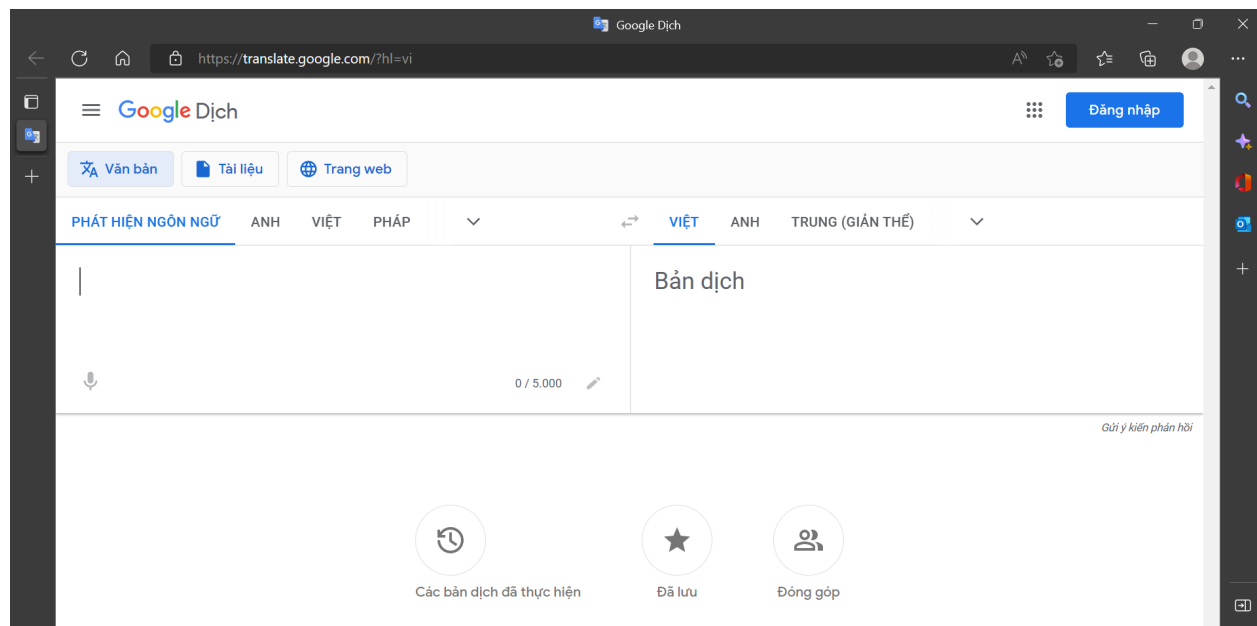
: Chức năng phát âm

3.2.2.b) Giao diện từ điển Việt-Anh.




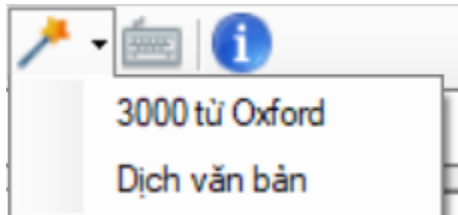
3.2.3. Google dịch

Sau khi click chuột trái 2 lần vào biểu tượng  ta sẽ được dẫn đến trang web của Google dịch.



3.2.4 Công cụ

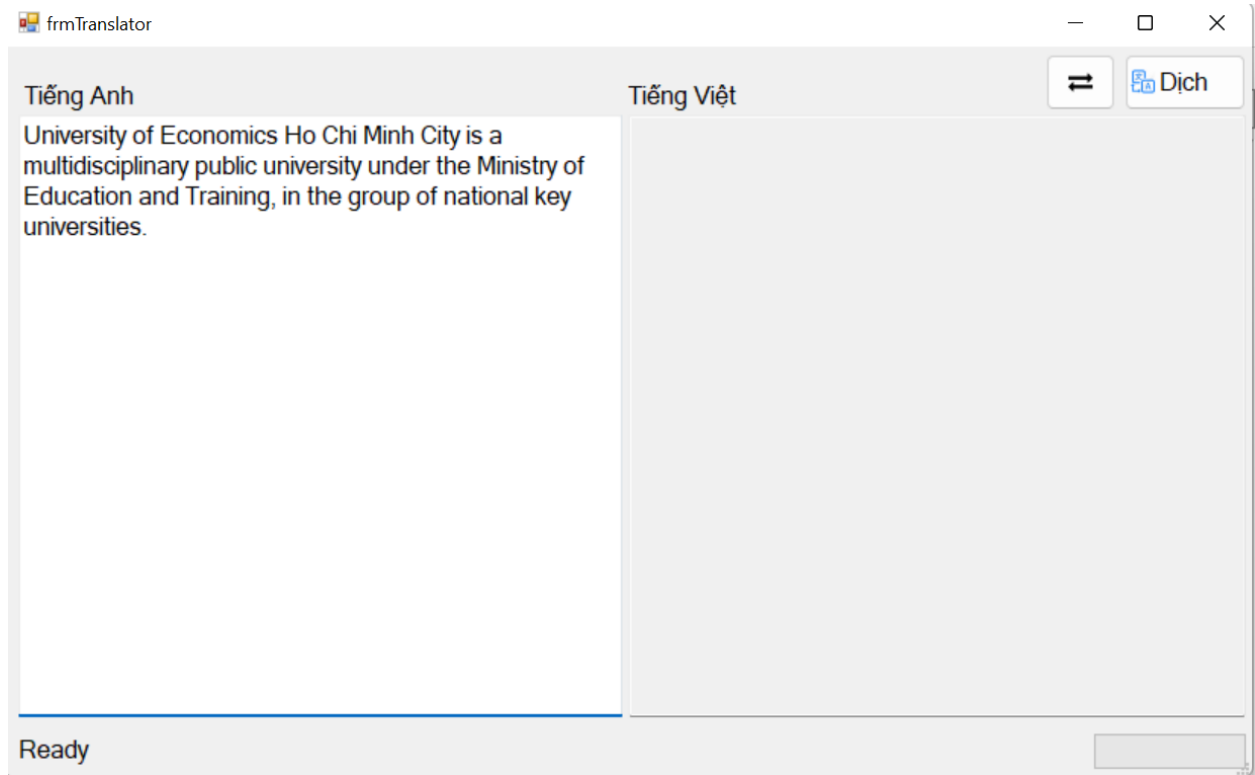
Khi di chuyển con chuột đến vị trí  và click vào biểu tượng trên ta sẽ nhận được 2 lựa chọn sau đây: 3000 Oxford và Dịch văn bản.



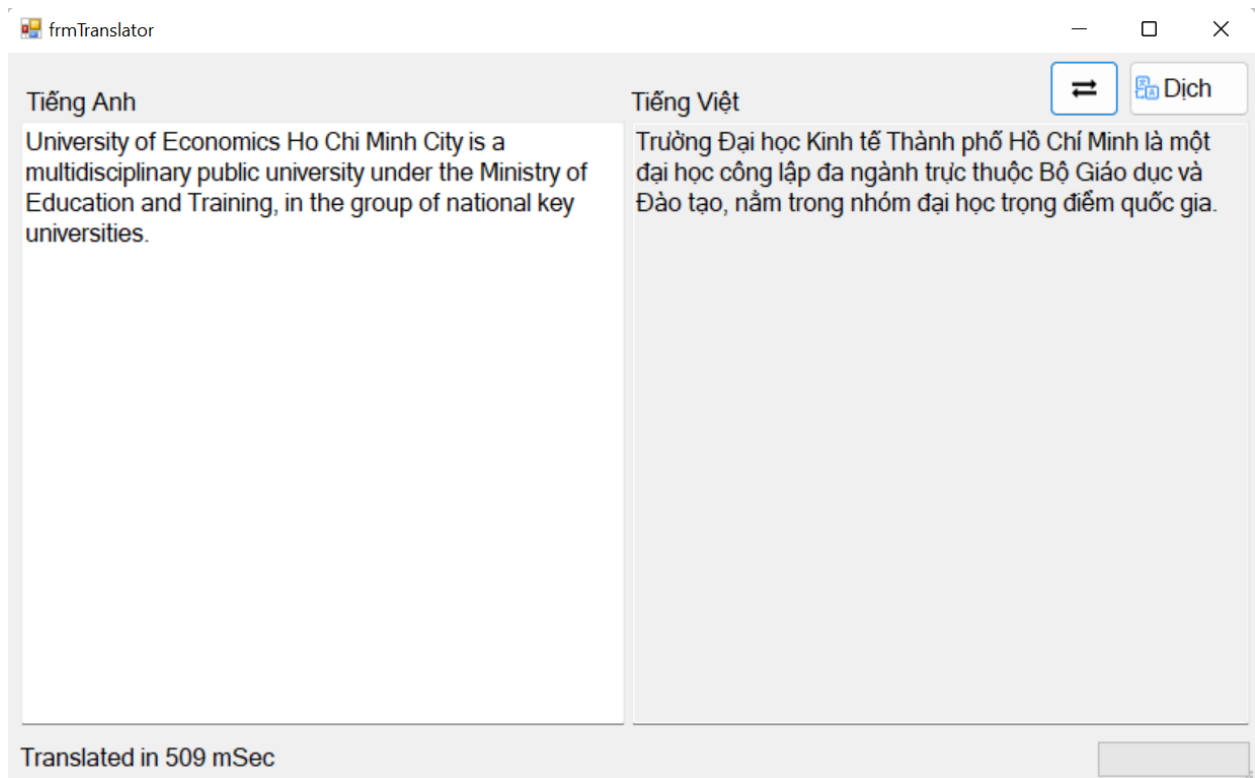
3.2.4.a) 3000 từ Oxford

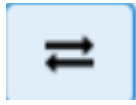


3.2.4.b) Dịch văn bản



Sau khi ghi đoạn văn bản cần dịch sang Tiếng Việt, click 2 lần chuột trái vào  để tiến hành dịch văn bản.





Ngoài ra có thể sử dụng chức năng

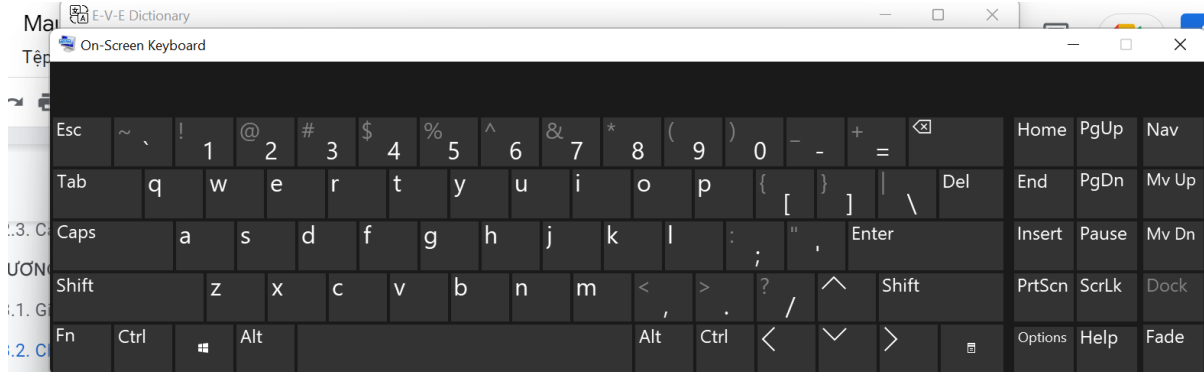
Translated in 509 mSec

Thể hiện khoảng thời gian dịch đoạn văn bản trên.

3.2.5. Bàn phím ảo



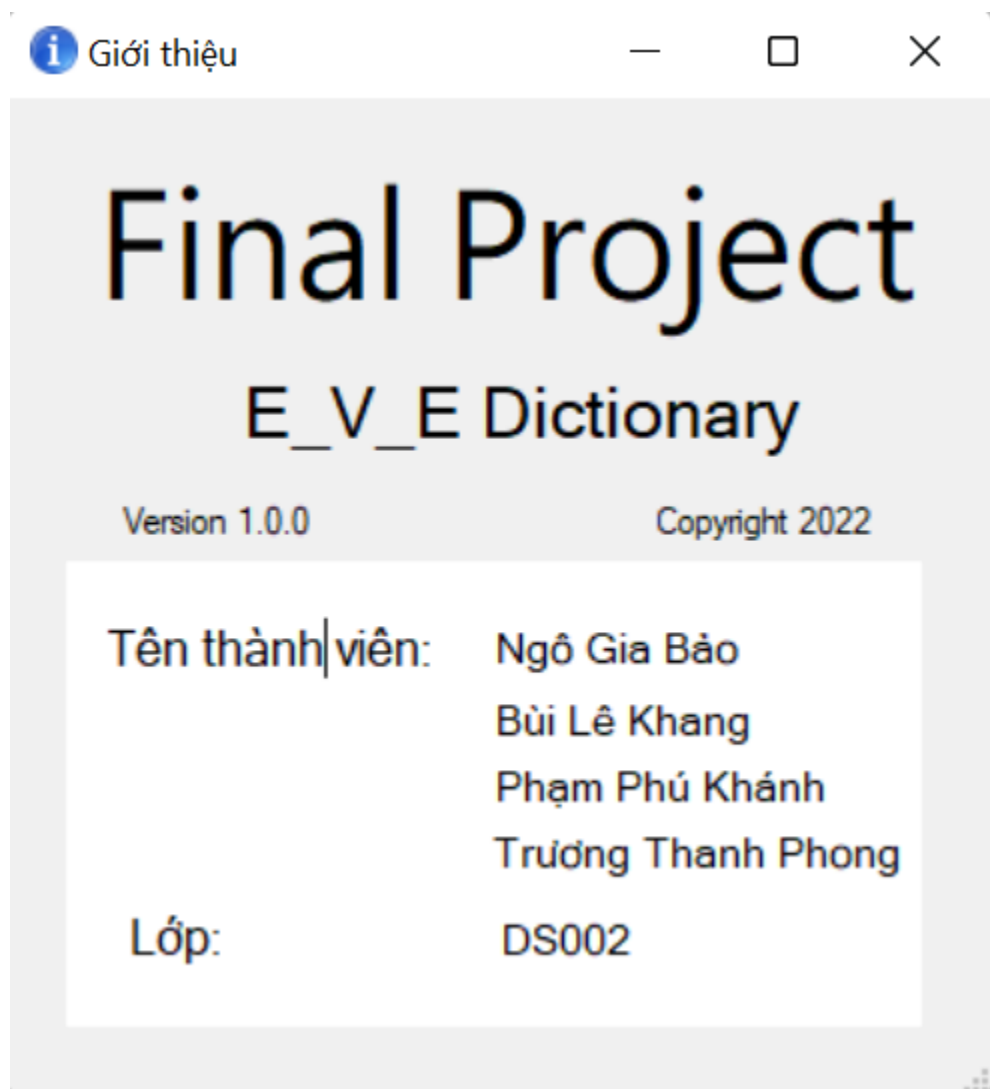
Click 2 lần chuột trái vào biểu tượng để sử dụng chức năng bàn phím ảo trong từ điển.



3.2.6. Giới thiệu ứng dụng



Click 2 lần vào để vào phần giao diện Giới thiệu dụng.

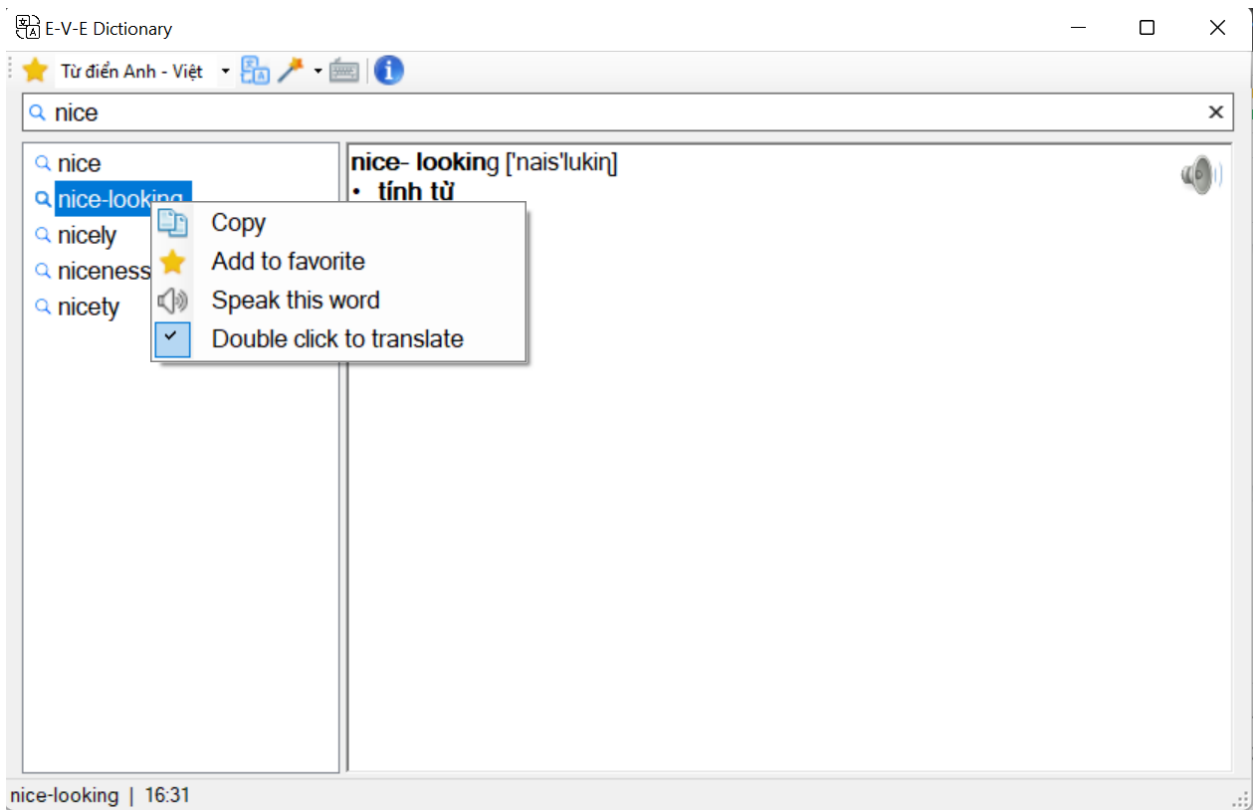


CHƯƠNG 4. THẢO LUẬN & ĐÁNH GIÁ

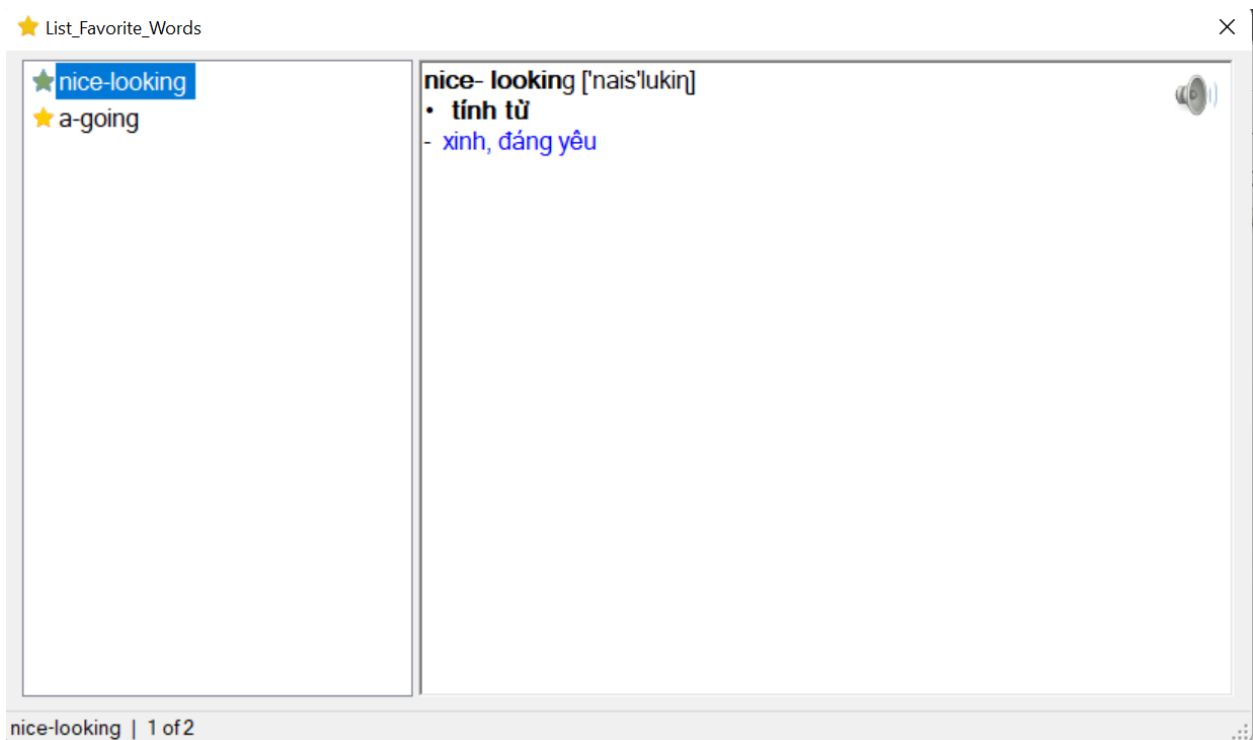
4.1. Các Kết Quả Nhận Được

4.1.1. Thêm từ vào Từ Yêu thích bằng thuật toán Add.

Click chuột phải vào từ muốn thêm vào Từ Yêu Thích

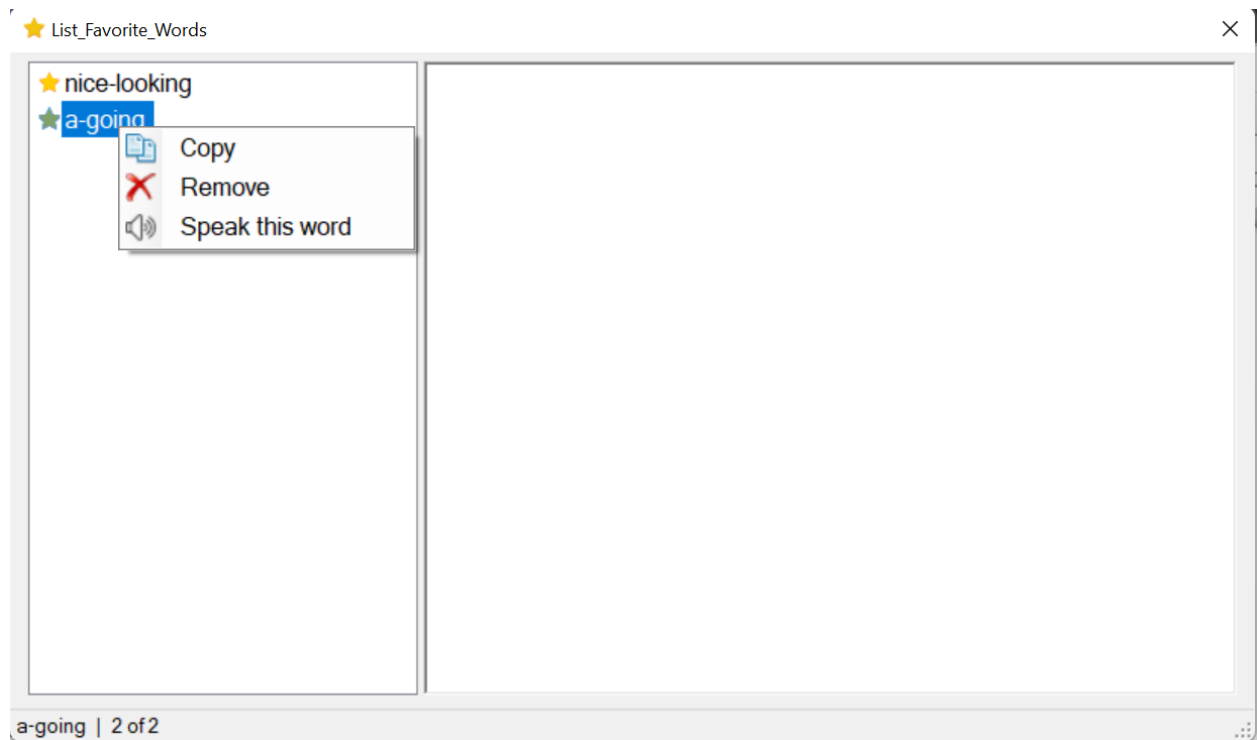


Chọn add to favorite, từ vừa chọn sẽ được thêm vào mục Từ Yêu Thích.

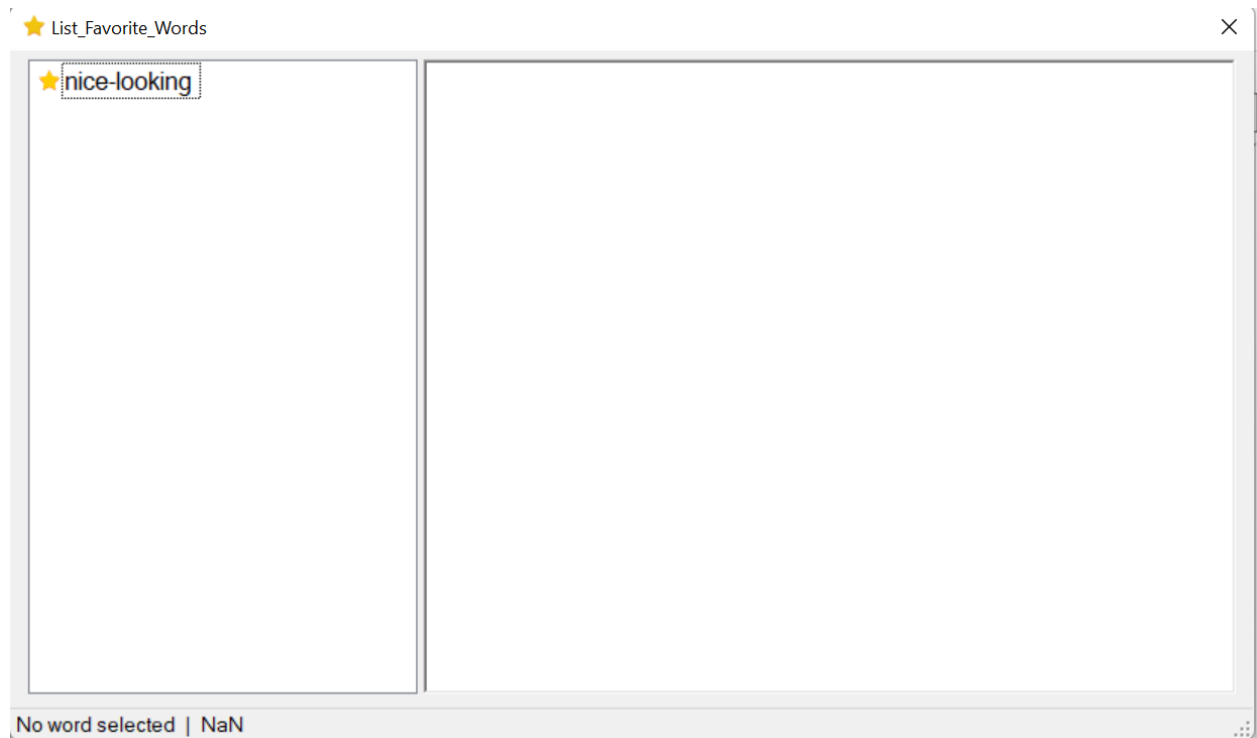


4.1.2. Xóa từ vựng khỏi Từ Yêu Thích bằng thuật toán Remove.

Click chuột phải vào từ vựng trong chức năng Từ Yêu Thích



Chọn remove để xóa từ khỏi chức năng Từ Yêu Thích.

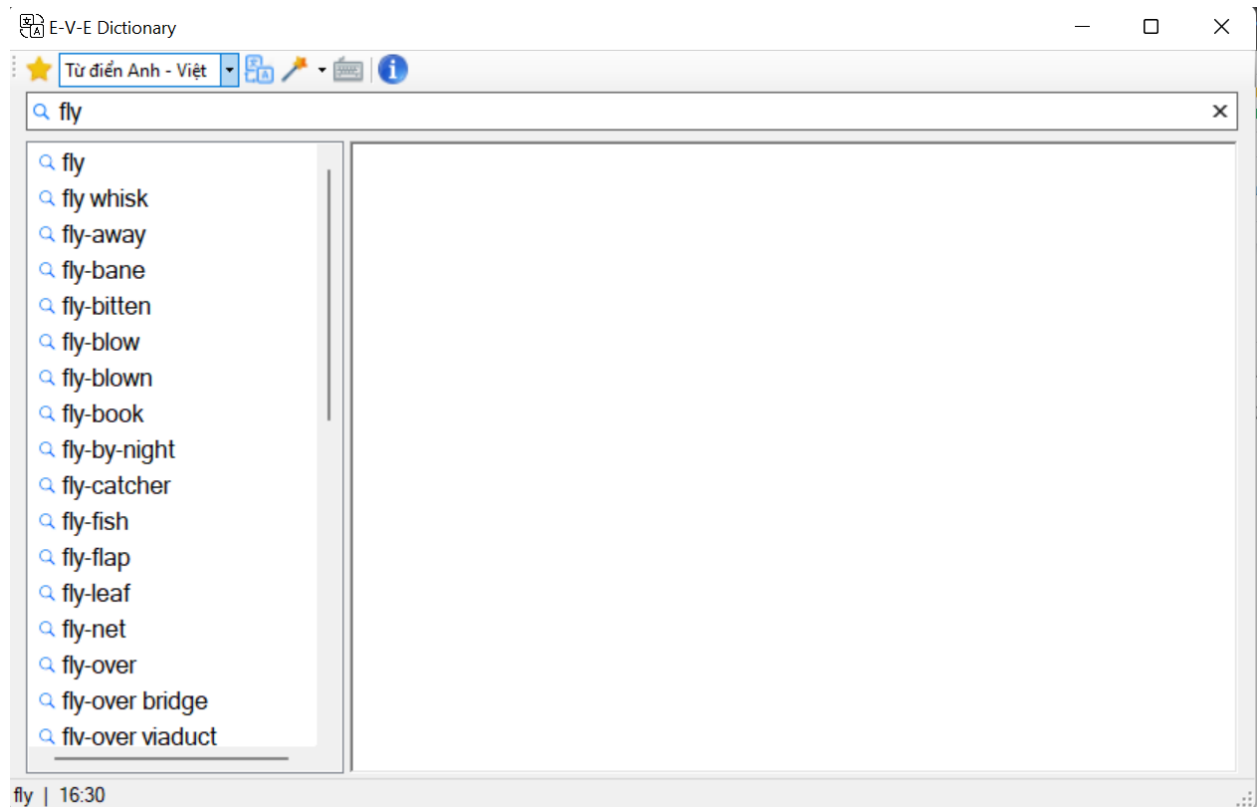


4.1.3 Tra từ vựng

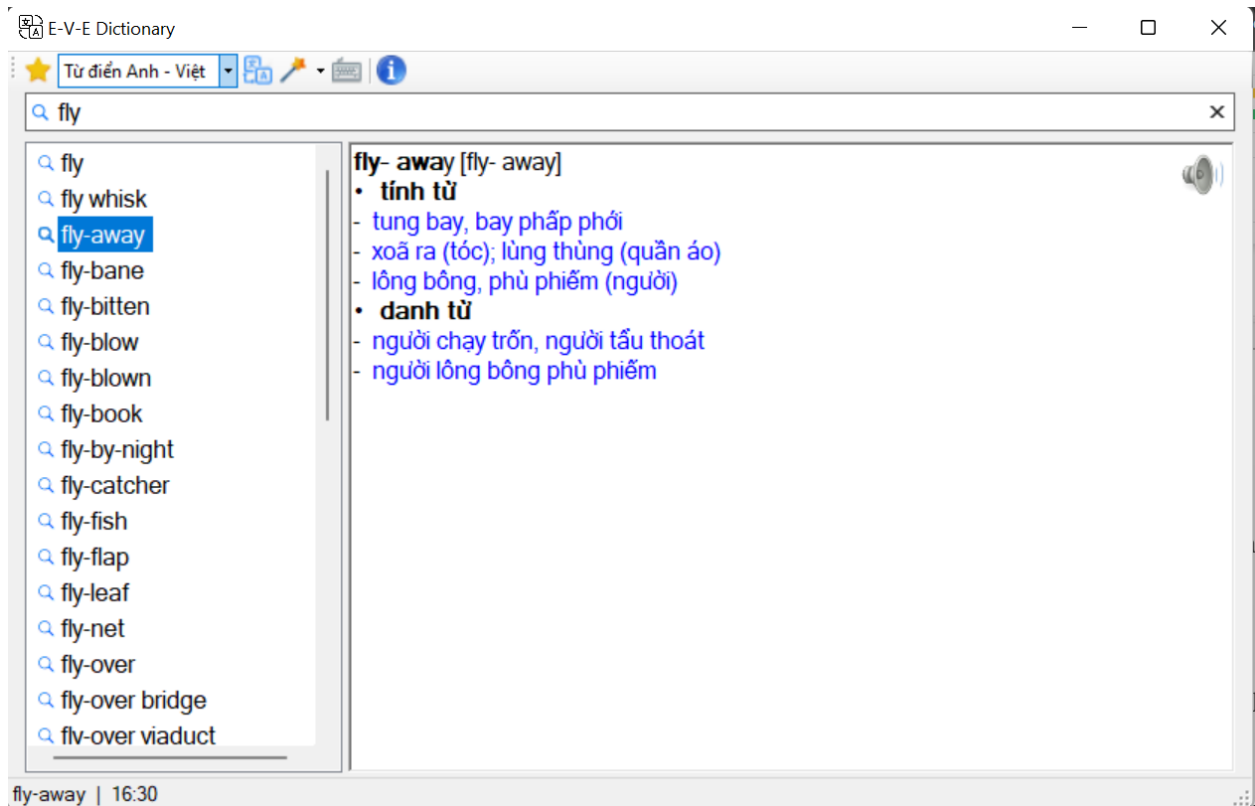
Tại phần tra cứu ở mỗi loại từ điển. Khi nhập một vài ký tự đầu tiên ngay lập tức bên



dưới sẽ gợi ý từ muốn tra cứu.



Sau đó chỉ cần click 2 lần chuột vào từ muốn tra cứu, ngay lập tức sẽ hiện lên loại từ, cách phát âm và nghĩa của từ.



4.2. Một Số Tồn Tại

Ngoài những ưu điểm trong việc tra cứu thì cấu trúc dữ liệu dictionary còn những vấn đề :

- Phải xác định trước cụ thể kiểu dữ liệu.
- Sẽ báo lỗi nếu Key để trống.
- Vì các phần tử được thêm theo thứ tự mà chúng ta thêm vào nên không thể tự động sắp xếp theo thứ tự bảng chữ cái nếu có một từ mới được thêm vào.
- Từ điển chỉ trả về giá trị value mà ở đây là những từ hoặc câu mà chúng ta tra cứu mà không biết những từ liên quan khác. Do đó, chúng ta sẽ gặp khó khăn khi nhớ một từ mới vì không thể áp dụng nó vào ngữ cảnh cụ thể.
- Do cấu trúc của từ điển là một cặp Key-Value nên nếu bạn nhập sai ngữ pháp thì sẽ không cho ra kết quả.
- Do có quá nhiều thông tin nên chiếm nhiều dung lượng bộ nhớ.
- Không phù hợp với tất cả đối tượng vì cấu trúc từ điển chỉ giải nghĩa theo mức độ cơ bản hoặc nâng cao.

4.3. Hướng Phát Triển

Bổ sung nhiều chức năng trong từ điển:

- ❖ Thêm nhiều ngôn ngữ khác như Hàn, Nhật, Trung và có thể dịch hai chiều.
- ❖ Bổ sung việc tra từ điển từ giọng nói.
- ❖ Có thể dịch đoạn văn bản từ hình ảnh.
- ❖ Tạo ra nhiều thẻ thông tin để học từ vựng từ danh sách các từ yêu thích.
- ❖ Thêm hình ảnh hoặc các video để minh họa cho các từ vựng hay tạo ra một số câu hỏi trắc nghiệm để ôn lại kiến thức.
- ❖ Thêm động từ bất quy tắc và những cấu trúc ngữ pháp quan trọng.

PHỤ LỤC

- Đưa Link Toàn Bộ Mã Nguồn Lên GitHub và Đưa Link Vào Đây

<https://github.com/BaoNgoDS/Final>

- Viết Hướng Dẫn Cách Cài Đặt Để Chạy

B1: Đảm bảo là máy đã có sẵn Visual Studio, và đã cài C#

B2: Chạy file English_Vietnamese Dictionary.sln

B3: Nhấn F5 để chạy chương trình

B4: Sử dụng ứng dụng.

- Phân Công Công Việc

Thành Viên	Nhiệm Vụ
Ngô Gia Bảo	1.2 Cấu trúc và cài đặt Dictionary. 1.3 Các thuật toán trên Dictionary. 3.1 Giao diện Menu chính. Viết hàm Add trong bài toán thêm từ vào Từ Yêu Thích.
Phạm Phú Khánh	1.1 Các khái niệm liên quan. 2.1 Phân tích bài toán. 3.2.1 Từ yêu thích. 3.2.5 Bàn phím ảo. 4.1 Các kết quả nhận được Tạo cơ sở dữ liệu Anh-Việt Việt-Anh, tạo lớp dictionary trong phần từ điển Anh-Việt Việt-Anh.
Bùi Lê Khang	3.2.4 Giới thiệu ứng dụng. 4.2 Một số tồn tại. 4.3 Hướng phát triển. Viết hàm Remove trong bài toán xóa từ khỏi Từ Yêu Thích.
Trương Thanh Phong	2.2 Sơ đồ lớp. 2.3 Cài đặt lớp. 3.2.2 Loại từ điển. 3.2.4 Công cụ. 3.2.3 Google dịch. Viết hàm tìm kiếm từ vựng và gợi ý từ vựng trong lúc tra từ.

TÀI LIỆU THAM KHẢO

1. <https://howkteam.vn/course/dictionary-trong-c/dictionary-trong-c-1572>
2. <https://www.geeksforgeeks.org/c-sharp-dictionary-with-examples>
3. <https://www.tutorialsteacher.com/csharp/csharp-dictionary>