AC12001: Week 8-11: Team Project

# Aim

This assignment is a team project where we have to use our knowledge of Java and data structures to design and implement a solution of a real-world problem. The aims of this project are to give us an opportunity to apply the knowledge we gained in AC11001 and AC12001. This will also give ous experience of working as a member of a team and we will develop our time management and our reporting skills.

The real-world problem we are tasked with is to develop a piece of software to perform language translation to and from English to one or more other languages.

We use the Yandex API to receive translation of words, which we use to build our own database. It is stored alongside the program. On runtime, the translations, or mapping of words, are stored in HashMaps.

# Requirements

### Functional Requirements:

1. The program shall translate individual words from English to German and from German to English.
2. The program shall provide a menu.
3. The program shall provide a settings option in the menu.
4. The program shall provide a setting that allows users to add new words to the translation dictionary when your translator cannot find the word in the dictionary.
5. The program shall take input from the user and translate it.
6. The program shall provide functions to save dictionaries to file and to load them from file.
7. The program shall provide an option to load and translate a text file.
8. The program shall output the translation time in words per second for the text file.
9. The program shall provide an option to remove a word or phrase from the dictionaries.
10. The program shall provide an option to display the dictionaries.
11. The program should run a GUI.
12. The program should add words from an API to the dictionary if the word doesn’t exist in the dictionary yet.
13. The program may detect the language of the input automatically.
14. The program may translate individual phrases.
15. The program may include an option to edit an entity in the dictionary.
16. The program may support the translations between additional languages.
17. The Program may include techniques to improve the quality of the translation such as ways to handle grammar differences between the translated languages.

### Non-functional Requirements:

1. The program shall include a jar file and JavaDoc.
2. The Java Runtime Environment (JRE) shall be installed on the system.
3. The program should translate at least 10 words per second.
4. The program should run on Windows and Linux.

# Use Cases

|  |  |  |
| --- | --- | --- |
| Translate a text from user input | | |
| 1 | User: | Selects the translate option in the menu |
| 2 | User: | Enters the text to translate |
| 3 | System: | Checks the language of the user input |
| 4 | System: | Translates the text to the corresponding language |
| 5 | System: | Displays the translated text to the user |

|  |  |  |
| --- | --- | --- |
| Load a text from file and translate it | | |
| 1 | User: | Selects the translate from file option in the menu |
| 2 | User: | Selects the file to be translated |
| 3 | System: | Loads selected file |
| 4 | System: | Checks the language of the text in the file |
| 5 | System: | Translates the text to the corresponding language |
| 6 | System: | Displays the translated text to the user |

# Class Definitions

|  |  |
| --- | --- |
| MenuChoice | |
| translate() | Calls the translate method from the Translator class |
| loadFileToTranslate() | Calls load file and translate from the Translator |

|  |  |
| --- | --- |
| Translator | |
| loadDictionaryFromFile() | Loads the dictionary from the file into the Dictionary class |
| saveDictionaryToFile() | Saves the complete dictionary to a file |
| loadFile() | Loads the contents of a file |
| translate() | Translates the text |

|  |  |
| --- | --- |
| Dictionary | |
| find() | Finds a word in the Dictionary |
| add() | Adds a new word to the Dictionary |
| remove() | Remove a word from the dictionary |
| toString() | Prints the whole Dictionary to screen |

# Pseudocode

translate (word, toGerman)

If (word in local dictionary)

return dictionary[“word”]

else

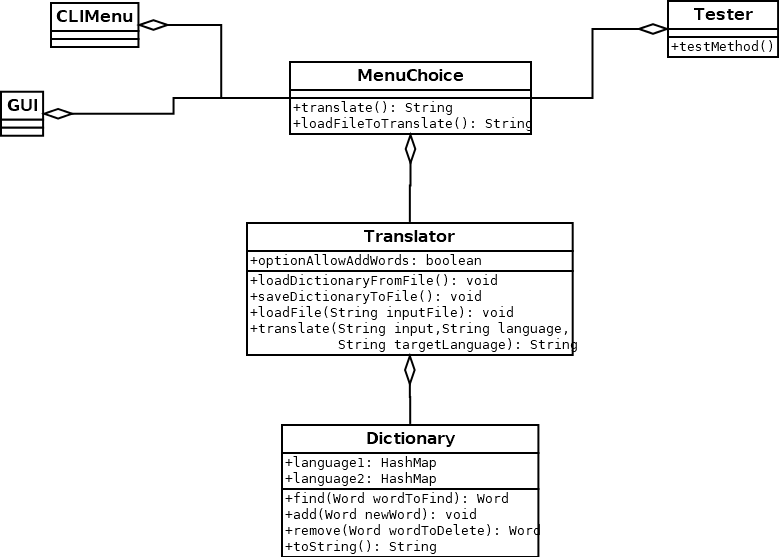
connect to online dictionary API

request translation

add translation to dictionary

return translation

# Class Diagram



# Tasks

|  |  |
| --- | --- |
| Alina (Secretary) | Design & Implementation: GUI, Tester, MenuChoice  Interim, Documentation, Test Plans, Evaluation of the project |
| Bjarne | Design & Implementation: Translator & Dictionary  Class Design, Evaluation of the project |
| Momchil | Design & Implementation: CLIMenu, Tester, MenuChoice  Presentation, Evaluation of the project |
| Panagiotis | Design & Implementation: Translator & Dictionary  Class Design, Evaluation of the project |