Name & ID:

Fatema Akter - 2222149642

Syed Sadat Adnan - 2232894642

Samin Shams Ahmed - 2014112042

Group No - 10

CSE 215.08

Submitted To: **Dr. Mohammad Shifat-E-Rabbi (MSRb)**

Project Report: Language Translation Tool.

Abstract

The **Language Translation Tool** is a software system designed to provide efficient, real-time translation services across different languages. The tool allows users to translate text between different languages, and it is designed to store translations in a properties file, ensuring easy retrieval and addition of new translations. Additionally, the system logs each translation along with a timestamp for record-keeping. This tool also offers the functionality to expand its translation database by accepting user-supplied translations, storing them for future use. The implementation relies on Java's built-in file handling capabilities to read and write translation data. The tool provides a simple and efficient solution to real-time language translation.

Introduction

Language barriers have always been a challenge in global communication. The **Language Translation Tool** is a solution to overcome this challenge by automating the translation process. It is designed to offer translations between different languages through a simple user interface. Built with Java, the tool processes translation requests sequentially and retrieves the required translations from a stored database.

The system provides a straightforward method of translating text, which can be extended by adding new translations dynamically. Additionally, it logs every translation request, ensuring that every interaction with the tool is recorded with a timestamp. This feature can be helpful for auditing and tracking the translations over time.

The application is particularly useful for people who need quick translations or those who work in environments requiring frequent text translations. It can be applied

in educational settings, travel, customer service, and anywhere cross-lingual communication is required.

Literature Review

Several studies and tools have been developed for language translation, ranging from rule-based systems to machine learning-based approaches. Google Translate, for instance, leverages neural networks to offer real-time translations between a wide variety of languages. While these systems are robust, they often come with high latency when used by multiple users at once or when integrated with real-time applications.

Many translation tools, however, lack customization and extensibility. They typically offer predefined translations that cannot be altered without substantial reconfiguration. Additionally, there is a need for a more efficient approach to improving translation response times without relying on complex systems.

The **Language Translation Tool** seeks to address these issues by offering a simple, extensible solution that not only translates predefined text but also allows users to add new translations dynamically. This feature sets it apart from many existing tools and makes it particularly useful for scenarios requiring frequent translation updates.

Methodology

The **Language Translation Tool** was developed using Java. It utilizes a Properties file to store translations in a key-value format. The following steps summarize the development process:

1. **Translation Database**: Translations are stored in a .properties file, with each translation corresponding to a key (the original text) and a value (the translated text). If a translation is not found, the system prompts the user to provide a new translation, which is then added to the database.
2. **File Handling**: Java's FileReader and FileWriter classes are used to read from and write to the translation properties file and the translation log file. This ensures that the tool can dynamically update and retrieve translations.
3. **Logging**: Every translation request, including both the original text and its translated version, is logged with a timestamp. This logging functionality is essential for tracking translation history and usage.
4. **User Interface**: The tool's interface is console-based, allowing users to select the language they want to translate to and then input text. The system will then process the input, display the translated text, and log the translation.

System Design

Architecture Overview

The tool follows a simple client-server architecture where the client is the user interacting through the console interface, and the server is the TranslationTool class that processes requests and handles translation data. The system architecture can be broken down into the following components:

- **User Interface**: The user inputs the text to be translated, and the system returns the translated text. If the translation is not available, the user can add a new translation, which is then saved to the file for future use.
- **Translation Engine**: This component is responsible for looking up existing translations and adding new ones. It uses a Properties file to store the translations.
- **Logging**: Every translation interaction is logged to a separate log file, which includes both the original text and its translation along with the time of the request.

Implementation

The system is implemented in Java, and it relies heavily on the Properties class for storing translations. The code is structured as follows:

1. **TranslationTool Class**:
    - Manages the loading and storing of translations.
    - Handles user input and translation requests.
2. **Main Class**:
    - Displays the user interface and prompts the user to select the translation language.
    - Takes user input and passes it to the TranslationTool for processing.
    - Handles the logic of adding new translations and logging each translation request.

Results

The **Language Translation Tool** was successfully implemented, providing the following features:

1. **Dynamic Translation Addition**: The tool allows users to add new translations to the database, making it adaptable and extensible.
2. **Translation Logging**: All translation requests are logged with a timestamp, which helps in tracking the history of translation interactions.
3. **Ease of Use**: The console interface is simple to navigate, and the tool provides clear prompts to guide users through the translation process.

Conclusion

The **Language Translation Tool** is an efficient and user-friendly translation tool that allows users to translate text between different languages. It offers dynamic translation capabilities, allowing users to add new translations to the database. The tool also ensures that all translation requests are logged for future reference. This system is particularly useful for environments requiring frequent translations and serves as a flexible, scalable solution for language translation.