Volume 1 Number 1 (2025)
January-April 2025
Page: 1-7

**JCIDP: Journal of Computational Intelligence anc
Data Processing**
https://edujavare.com/index.php/JCIDP

Implementation of Object-Oriented Programming in a Website-Based "Dictionary Translate" Application Using Python

Kevin Forggan Coal[1], Rafli Sultan Pradana S[2]

[1] State University of Medan, Indonesia

* Correspondence e-mail; kevinbatubara355@gmail.com

Article history Submitted: 2025/03/01; Revised: 2025/03/11; Accepted: 2025/04/22

Abstract

The development of a web-based translator application using the Python language aims to help users translate text into various languages practically and efficiently. This application was developed using the Tkinter library to build an interactive graphical interface as well as the GoogleTrans module for the translation process without relying on external APIs. With the login feature, this application provides safer access control for users. Apart from that, users can select various destination languages according to their needs through the menu provided. The use of Python in developing this application offers ease of maintenance and flexibility in developing additional features. With a simple appearance and easy to use, this application can be an alternative solution for users who need a fast translation tool without having to connect to cloud-based services.

Keywords Text Translator, Python, Tkinter, googletrans, Graphical Interface

© 2025 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution 4.0 International (CC BY SA) license, https://creativecommons.org/licenses/by-sa/4.0/.

INTRODUCTION

The development of information and communication technology in the modern era has brought significant changes in various aspects of human life. Rapid advances in the fields of software and artificial intelligence have enabled the development of various applications that support productivity and make daily activities easier. One area experiencing rapid development is language translation technology, which is increasingly needed in various sectors, including education, business and global communications.

The use of computer and internet-based devices has now become a major need for society. Devices such as smartphones, computers and laptops have replaced manual methods in various fields of work, including cross-language communication. However, many translation applications available today still rely on cloud-based services or require an internet connection to function optimally. This is an obstacle for users who need translation services offline or without stable network access. Therefore, an alternative solution is needed in the form of a translation application that can work independently without relying on cloud services. Software development demands development methods that are flexible, efficient, and easy to use. One approach that is often used in modern software development is object-based programming, which allows code management to be more modular, reusable, and makes further development easier. In this context, Python has become a very popular programming language because of its simple syntax, ease of developing GUI-based applications, and extensive library support.

In this research, a web-based translation application using the Python language was developed which aims to provide offline text translation services with an interactive and easy-to-use interface. This application was built using the Tkinter library for graphical interface development and the GoogleTrans module for the translation process. One of the main features implemented is a login system, which functions to limit user access and increase security in using the application.

With the presence of this Python-based translator application, it is hoped that it can provide a more practical, efficient and cost-effective alternative for users who frequently interact with various languages but have limited internet access. In addition, this application can be used by various groups, including students, working professionals, and researchers who need an instant translation tool that is easy to use. In the future, further development can be carried out by adding machine learning features to increase translation accuracy and expand the range of supported languages.

METHODS

This research uses the Research and Development (R&D) method, which is an approach that aims to develop and test an application so that it can be used effectively. In this research, the product developed is a web-based translation application using Python and Tkinter which allows users to translate text without relying on cloud-based services. To develop this application, the Waterfall software development model is used which consists of several stages, namely requirements analysis, design, implementation, testing and maintenance. The needs analysis stage is carried out by identifying the features needed in a translator application, including support for various languages and integration of translation modules. Next, the design stage includes designing the user interface (UI/UX) using Tkinter as well as determining the data structure and translation algorithm. After the design is complete, the implementation stage is carried out by writing program code based on the design that has been created, using Python libraries such as Googletrans for the translation process.

The next stage is testing, which is carried out to ensure the application functions as expected. Testing is carried out using the Black Box Testing method, namely by checking whether all features work correctly without looking at the source code, as well as Usability Testing, which involves users in assessing the ease of use of the application. In addition, data is collected through observation, interviews, and experiments to understand user needs and test translation accuracy and speed. If necessary, the Prototype model can be used as an alternative in the development process by building an initial prototype, collecting input from users, and iterating on improvements before releasing the final version.

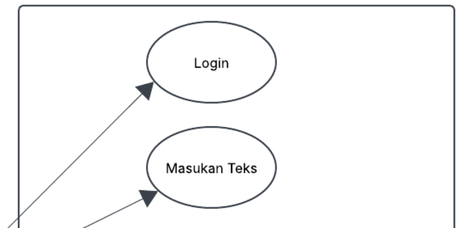
FINDINGS AND DISCUSSION

In this article, the author develops a web-based translator application using the Python language with the aim of making it easy to translate text into various languages efficiently. This application was developed using the Tkinter library to build an interactive graphical interface and the GoogleTrans module for the translation process without dependence on external APIs. In developing this application, a mechanism was needed that could handle text input and output in real-time. Therefore, the system is designed with a modular structure, where each component has a specific function, such as user interface, text processing, as well as translation.

The author uses Python as the main programming language because it is flexible, easy to understand, and supports various libraries that can be used for text processing and graphical interfaces.

a. Use Case Diagram

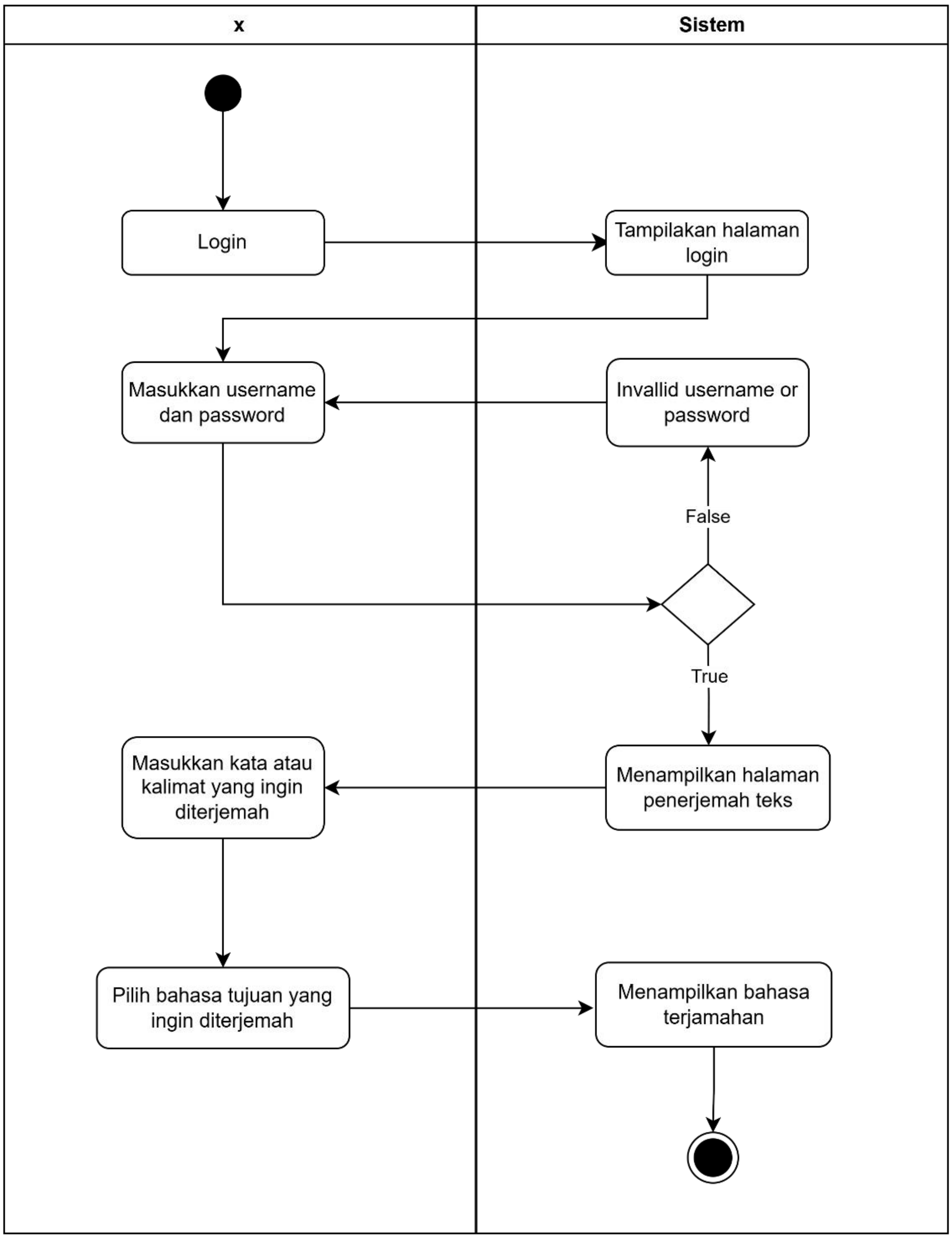
Use Case Diagram is a type of diagram in the Unified Modeling Language (UML) which is used to describe relationships and interactions between actors and systems. This diagram shows how the user interacts with the system through the various scenarios or functions available. Use Case Diagrams also play a role in describing the types of interactions that users can carry out with the system, thereby helping to understand the needs and behavior of the system as a whole.



b. Activity Diagram

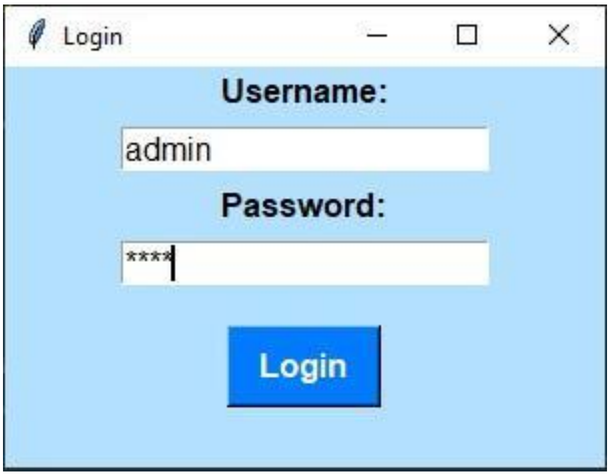
Activity diagrams, or activity diagrams in Indonesian, are diagrams used to model various processes that occur in a system. This diagram depicts the workflow sequence of a system visually and structured, usually in a vertical orientation. As a type of diagram in the Unified Modeling Language (UML), activity diagrams function as a further development of Use Case Diagrams, helping to understand how a process or activity takes place in the system in more detail.

`


x
Sistem
Login
Tampilakan halaman login
Masukkan username dan password
Invallid username or password
False
True
Masukkan kata atau kalimat yang ingin diterjemah
Menampilkan halaman penerjemah teks
Pilih bahasa tujuan yang ingin diterjemah
Menampilkan bahasa terjamahan

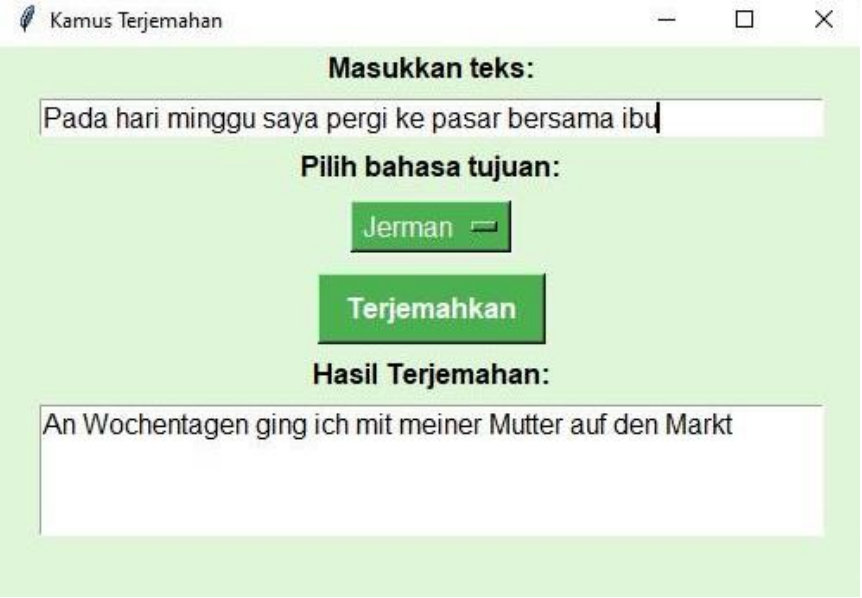
c. Running the Program and Implementing the GUI

In the first stage, the system will display a login menu. In the login menu, the user must enter the username and password correctly to be able to enter the main menu page.



Next, from the login menu the user will be directed to the main menu after entering the username and password correctly. If the username or password is incorrect, the system will validate and send a message that the username or password entered is incorrect.

In the main menu display, users can input text in the form of words or sentences that they want to translate. Then the user can then select the language destination they want to translate. An example of what it looks like is in the image below:



Masukkan teks: Pada hari minggu saya pergi ke pasar bersama ibu

Pilih bahasa tujuan: Jerman

Hasil Terjemahan: An Wochentagen ging ich mit meiner Mutter auf den Markt

CONCLUSION

The development of a web-based translator application using the Python language is that this application is able to make it easy for users to translate text into various languages quickly and efficiently. By utilizing the Tkinter library, this application has an interactive and easy-to-use graphical interface without requiring a connection to cloud-based services or external APIs. In addition, the implemented login feature increases access security for users, while flexible destination language selection allows translation into various languages as needed. The use of Python in developing this application provides advantages in terms of flexibility and ease of maintenance, so that the application can be further developed with additional features in the future. With a simple appearance and ease of use, this application can be an alternative solution for users who need a practical and reliable offline translation tool.

REFERENCES

M. M. Shabrina, A. Aranta, and B. Irmawati, "DESIGN AND DESIGN OF AN INDONESIAN VOICE CONVERSION ALGORITHM TO LATIN TEXT IN SASAK USING DICTIONARY BASED METHOD (Design and Development of Indonesian Voice Conversion Algorithm to Sasak Language Latin Text Using Dictionary Based Method)," vol. 6, no. 1, pp. 364–375, 2024.

N. K. R. Sari, I. M. A. D. Suarjaya, and P. W. Buana, "Comparison of Translation Libraries in Python (Case Study: Sentiment Analysis of Infectious Diseases in Indonesia)," *J. Ilm. Technol. and Comput.*, vol. 2, no. 3, pp. 1–7, 2021.

R. Adolph, "CompletedNo Title No Title No Title," pp. 1–23, 2016.

A. A. Ghirrid, R. T. K. Sari, and R. T. Aldisa, "Natural Language Processing Algorithm for Translator Applications (Indonesia-Java) Using Speech Processing Methods," *J. JTIK (Journal...*, vol. 8, no. 3, 2024, [Online]. Available: https://journal.lembagakita.org/jtik/article/view/2244%0Ahttps://journal.lembagakita.org/jtik/article/download/2244/1991

M. Detriasmita, N. Wayan, I. G. Agung, and G. Arya, "Implementation of a Python-Based Multi-Language Translator Application with Integration of Google Translate API and Tkinter GUI," vol. 17, pp. 148–157, 1979.