Polyglot ■■■ – Piratical PyScript Translator

Abstract

This project demonstrates a lightweight web-based application that translates English text into “Pirate Speak” using PyScript and the arrr Python library. The project integrates HTML, PyScript, and Python to showcase client-side execution of Python code within the browser. Users can input English text, which is instantly converted to Pirate slang through a simple interface. The system highlights how PyScript can bridge Python with modern web development.

Introduction

Background: With the growth of web applications, Python developers often need tools to run Python code in browsers without relying on servers. PyScript offers this functionality by embedding Python directly into HTML.

Importance: This project demonstrates how to combine Python + Web (HTML/JS) technologies, lowering the barrier for beginners and showcasing real-world potential for lightweight applications.

Objectives:

- Build a web app that translates English into Pirate language.
- Implement PyScript for running Python directly in the browser.
- Demonstrate integration of external Python libraries (arrr).

Literature Review: Traditionally, such projects rely on JavaScript-based translators or server-side frameworks like Flask or Django. This project bypasses the need for servers by using PyScript.

System Requirements

Hardware:

- 4 GB RAM minimum
- Dual-core CPU
- Any modern browser

Software:

- Python library: arrr
- PyScript runtime
- IDE: VS Code, PyCharm
- OS: Windows/Linux/macOS

Methodology / Project Design

Approach:

1. Create a web interface (index.html).
2. Configure PyScript (pyscript.json).
3. Implement the translation function (pythonscript.py).
4. Bind button event using py-click.

Implementation

Modules Used:

- arrr: Pirate-speak translation.
- pyscript.document: DOM manipulation.

Code Files:

- index.html: Frontend structure.
- pyscript.json: Declares required packages.
- pythonscript.py: Defines translation function.

Results & Discussion

The project successfully converts English phrases into Pirate language within a browser.

Example:

Input: Hello friend, welcome aboard!
Output: Ahoy matey, welcome aboard!

This confirms PyScript can load Python modules and interact with HTML elements seamlessly.

Conclusion

Summary:

- Implemented a Python-based web app without a backend server.
- Demonstrated integration of PyScript + external Python libraries.
- Successfully translated English into Pirate slang.

Future Scope:

- Extend translation to more languages.
- Add speech features.
- Improve UI/UX with animation

References / Bibliography

- PyScript Documentation: https://pyscript.net
- arrr Python Library: https://pypi.org/project/arrr/
- Mozilla Developer Docs: https://developer.mozilla.org