Multilingual Learning Web Application

0. Introduction

0.1 Preface

This project originated from my personal experience with language learning. I was born and raised in Korea, and later continued my studies in France and the United States. Throughout this journey, I had to master two foreign languages to the level of understanding university-level coursework, which allowed me to deeply empathize with the struggles of language learners.

Today, with smartphone apps, it's easy to look up unfamiliar words. However, when I first started my studies in France, I often relied on paper or electronic dictionaries, searching for words one by one. Beyond grammar and vocabulary, I also had to study idiomatic expressions, synonyms, antonyms, and more. Looking up each word and writing it down to memorize it was not only time-consuming but also laborious. Moreover, the accuracy of online translation tools at the time was questionable, and reading just a few pages of a book could take half a day.

When I moved to the United States to study English again, I faced another challenge. While both French and English use the same alphabet, their pronunciation systems are completely different, and sometimes words that look similar have entirely different meanings. Using both languages simultaneously became confusing. As I focused more on English, my French skills naturally began to decline. Each time this happened, I tried to write and memorize both languages in my notes. But that process was exhausting and cumbersome, eventually leading me to give up. During those times, I often wished there had been a book that could help me learn both languages at the same time.

Remarkably, today, I study languages in a way that is vastly different from when I first started my studies abroad. Instead of referencing paper or electronic dictionaries to find individual words, I can now simply point to a word in an image and instantly discover its meaning. Additionally, tools like ChatGPT allow me to seamlessly translate entire pages of text into

Korean, French, and English, while also capturing the nuances of the original content. In just seconds, I can extract words, idiomatic expressions, and example sentences from lengthy passages, making the language learning process far more efficient. I believe this approach not only benefits those learning multiple languages simultaneously, but also individuals preparing for language proficiency exams such as TOEIC or IELTS.

With this background, I began exploring ways to leverage advanced technologies to make language learning more efficient and accessible. This project represents the first step toward realizing that vision, focusing not only on language translation, but also on providing tools that help learners maximize their language skills while minimizing the time and effort involved. Through this endeavor, I hope to offer tangible support to language learners worldwide, helping them to discover new opportunities and broaden their horizons.

0.2 Abstract

This project is grounded in my personal experience with foreign language acquisition, aiming to develop a web application that facilitates the efficient learning of multiple languages simultaneously. Traditionally, language learners have relied on paper or electronic dictionaries to look up individual words, a process that is both time-consuming and inefficient. However, the advent of AI-powered tools, such as ChatGPT, has ushered in a paradigm shift in language learning methodologies. These tools offer rapid and accurate translations, along with real-time suggestions for words, idiomatic expressions, and example sentences, thus enabling a more intuitive and efficient learning experience. As a result, learners can significantly reduce the time spent searching for and verifying vocabulary, improving overall learning efficiency. This project seeks to leverage these technological advancements to address the inherent inefficiencies of traditional language learning approaches. By integrating AI-driven solutions, the application aims to create an environment where learners can study multiple languages simultaneously, minimizing the time and effort required while maximizing learning outcomes.

---

1. Technology Stack

1.1 Development Tools

1) **Version Control Platform:** GitHub
2) **IDE:** Visual Studio Code

3) **Design:** Figma

   Chosen for its robust collaborative design capabilities, enabling real-time prototyping and efficient UI/UX development.

1.2 Core Technology

1) **Language:** Python
2) **Backend Framework:** Django

   Ensures secure, scalable, and efficient backend development with built-in database management.

3) **UI Framework:** React.js

1.3 Data & AI Services

1) **Dictionary:** Oxford Dictionaries API

   Provides accurate definitions, synonyms, and examples to support language learning

2) **Machine Translation:** DeepL API, Google Cloud Translation API

   Delivers high-quality, context-aware translations for effective multilingual communication

3) **Artificial Intelligence:** OpenAI API (GPT-4), Google Cloud AI

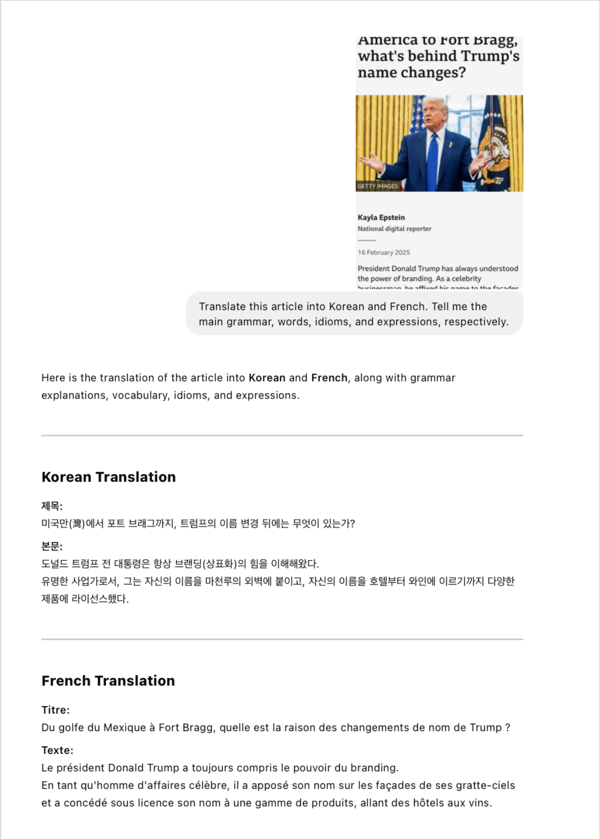
   Enables powerful AI-driven language processing and real-time text analysis in a web environment.

---

2. Methodology

2.1 Project Motivation & Background

This project was inspired by my experience of using AI tools to study multiple languages simultaneously. Through this process, I recognized the potential of AI in streamlining language learning by providing instant translations, contextual word meanings, grammar explanations, and usage examples. Below is a glimpse of how I utilize ChatGPT to retrieve the learning information I need.



America to Fort Bragg, what's behind Trump's name changes?

GETTY IMAGES

Kayla Epstein
National digital reporter

16 February 2025

President Donald Trump has always understood the power of branding. As a celebrity businessman, he affixed his name to the facades

Translate this article into Korean and French. Tell me the main grammar, words, idioms, and expressions, respectively.

Here is the translation of the article into **Korean** and **French**, along with grammar explanations, vocabulary, idioms, and expressions.

Korean Translation

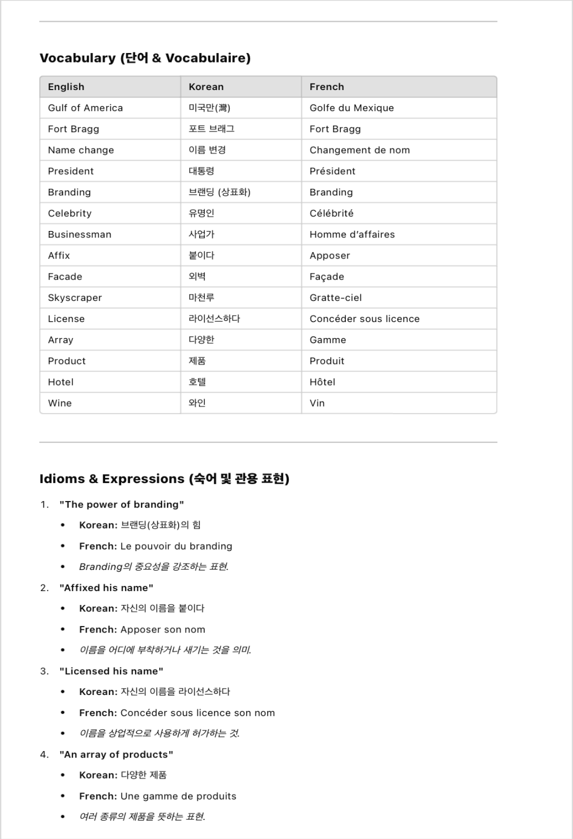
제목:
미국만(灣)에서 포트 브래그까지, 트럼프의 이름 변경 뒤에는 무엇이 있는가?

본문:
도널드 트럼프 전 대통령은 항상 브랜딩(상표화)의 힘을 이해해왔다.
유명한 사업가로서, 그는 자신의 이름을 마천루의 외벽에 붙이고, 자신의 이름을 호텔부터 와인에 이르기까지 다양한 제품에 라이선스했다.

French Translation

Titre:
Du golfe du Mexique à Fort Bragg, quelle est la raison des changements de nom de Trump ?

Texte:
Le président Donald Trump a toujours compris le pouvoir du branding.
En tant qu'homme d'affaires célèbre, il a apposé son nom sur les façades de ses gratte-ciels et a concédé sous licence son nom à une gamme de produits, allant des hôtels aux vins.

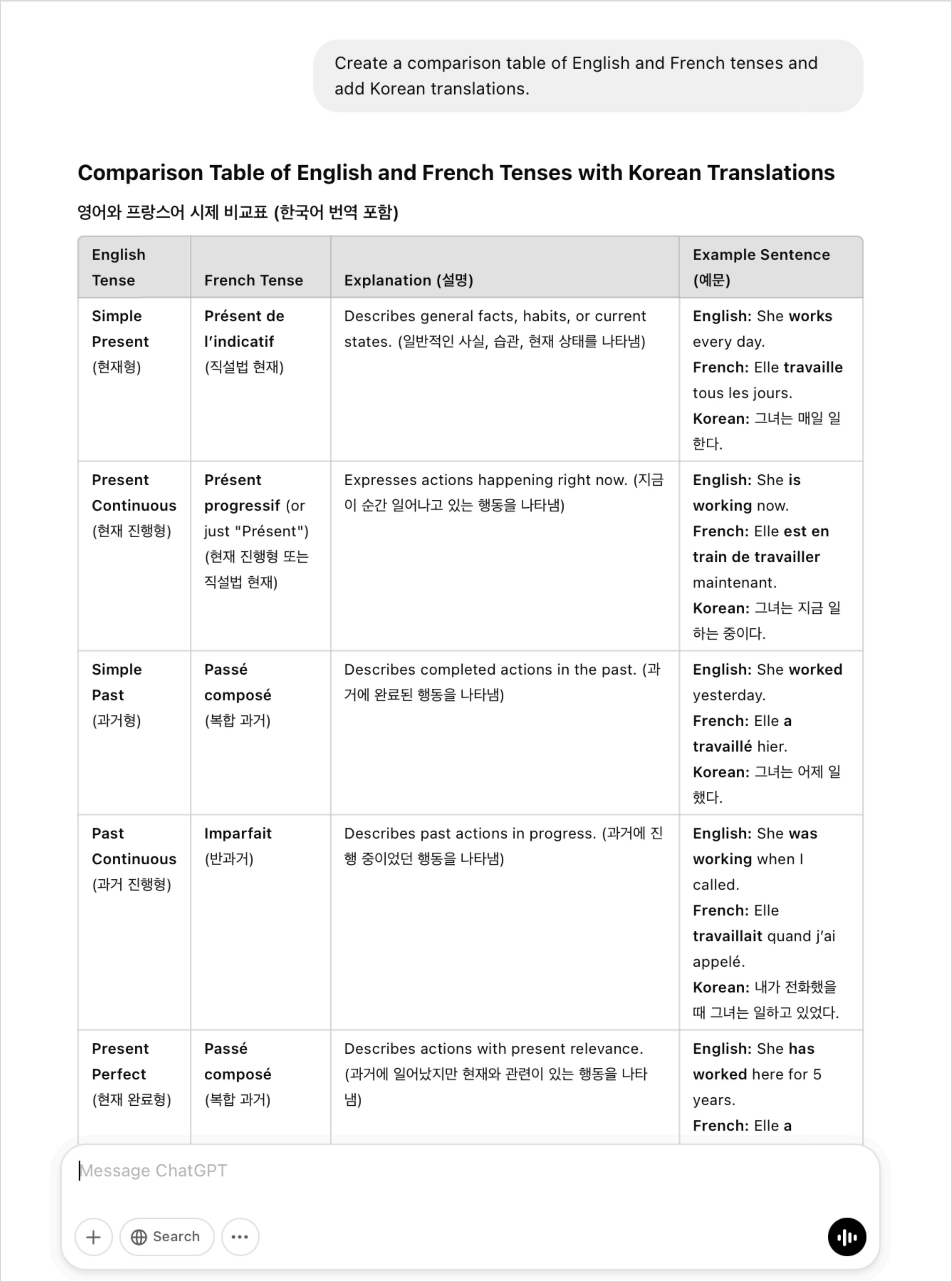


Vocabulary (단어 & Vocabulaire)

English	Korean	French
Gulf of America	미국만(灣)	Golfe du Mexique
Fort Bragg	포트 브래그	Fort Bragg
Name change	이름 변경	Changement de nom
President	대통령	Président
Branding	브랜딩 (상표화)	Branding
Celebrity	유명인	Célébrité
Businessman	사업가	Homme d'affaires
Affix	붙이다	Apposer
Facade	외벽	Façade
Skyscraper	마천루	Gratte-ciel
License	라이선스하다	Concéder sous licence
Array	다양한	Gamme
Product	제품	Produit
Hotel	호텔	Hôtel
Wine	와인	Vin

Idioms & Expressions (숙어 및 관용 표현)

1. **"The power of branding"**
   - **Korean:** 브랜딩(상표화)의 힘
   - **French:** Le pouvoir du branding
   - *Branding의 중요성을 강조하는 표현.*
2. **"Affixed his name"**
   - **Korean:** 자신의 이름을 붙이다
   - **French:** Apposer son nom
   - *이름을 어디에 부착하거나 새기는 것을 의미.*
3. **"Licensed his name"**
   - **Korean:** 자신의 이름을 라이선스하다
   - **French:** Concéder sous licence son nom
   - *이름을 상업적으로 사용하게 허가하는 것.*
4. **"An array of products"**
   - **Korean:** 다양한 제품
   - **French:** Une gamme de produits
   - *여러 종류의 제품을 뜻하는 표현.*



Create a comparison table of English and French tenses and add Korean translations.

Comparison Table of English and French Tenses with Korean Translations

영어와 프랑스어 시제 비교표 (한국어 번역 포함)

English Tense	French Tense	Explanation (설명)	Example Sentence (예문)
Simple Present (현재형)	**Présent de l'indicatif** (직설법 현재)	Describes general facts, habits, or current states. (일반적인 사실, 습관, 현재 상태를 나타냄)	**English:** She **works** every day. **French:** Elle **travaille** tous les jours. **Korean:** 그녀는 매일 일한다.
Present Continuous (현재 진행형)	**Présent progressif** (or just "Présent") (현재 진행형 또는 직설법 현재)	Expresses actions happening right now. (지금 이 순간 일어나고 있는 행동을 나타냄)	**English:** She **is working** now. **French:** Elle **est en train de travailler** maintenant. **Korean:** 그녀는 지금 일하는 중이다.
Simple Past (과거형)	**Passé composé** (복합 과거)	Describes completed actions in the past. (과거에 완료된 행동을 나타냄)	**English:** She **worked** yesterday. **French:** Elle **a travaillé** hier. **Korean:** 그녀는 어제 일했다.
Past Continuous (과거 진행형)	**Imparfait** (반과거)	Describes past actions in progress. (과거에 진행 중이었던 행동을 나타냄)	**English:** She **was working** when I called. **French:** Elle **travaillait** quand j'ai appelé. **Korean:** 내가 전화했을 때 그녀는 일하고 있었다.
Present Perfect (현재 완료형)	**Passé composé** (복합 과거)	Describes actions with present relevance. (과거에 일어났지만 현재와 관련이 있는 행동을 나타냄)	**English:** She **has worked** here for 5 years. **French:** Elle **a**

Figure1. An example of my language learning using ChatGPT, which inspired the project

2.2 System Flowchart

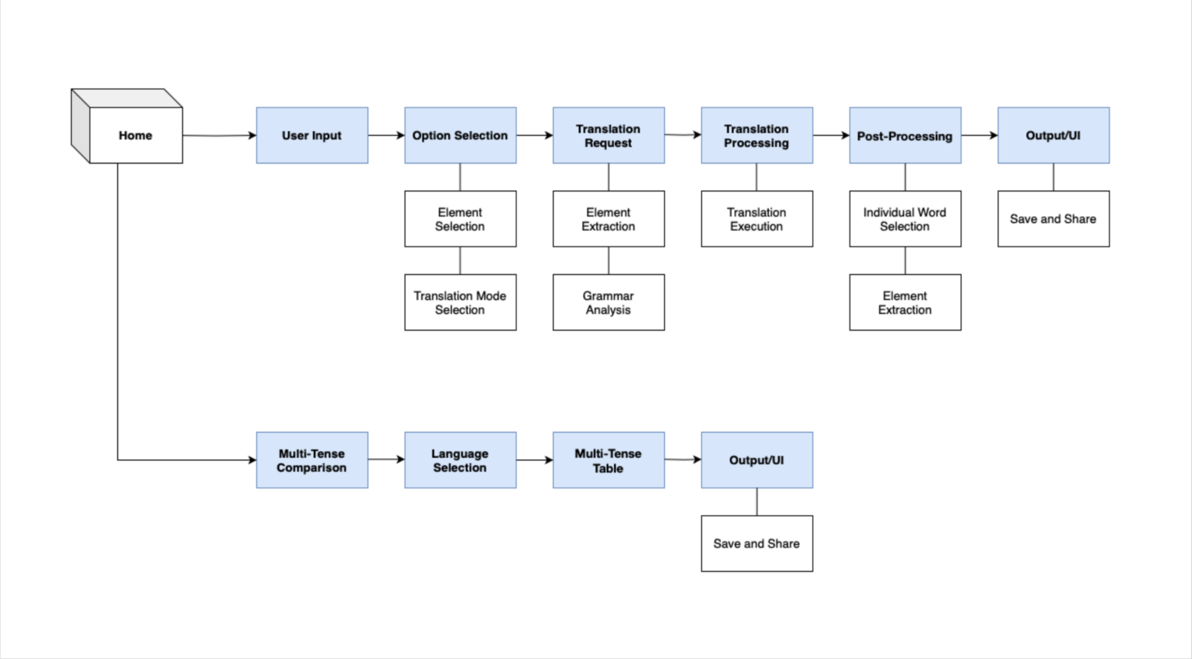


Figure 2. Multilingual Translation Web Application System Flowchart

This diagram presents the fundamental processes of a multilingual translation web application, which is structured into two primary workflows. The first process is the **general translation workflow**, which begins when the user inputs text and selects translation parameters. In this phase, users can specify translation elements, including words, idioms, as well as choose the translation mode—either simultaneous or separate translation. The system subsequently conducts text analysis, extracts relevant linguistic elements, and performs grammatical analysis before executing the translation. Following this, a post-processing phase enables users to refine the translation by selecting specific words or adjusting synonyms and antonyms. The final output is then presented through the user interface (UI), where users have the option to save or share the translated content.

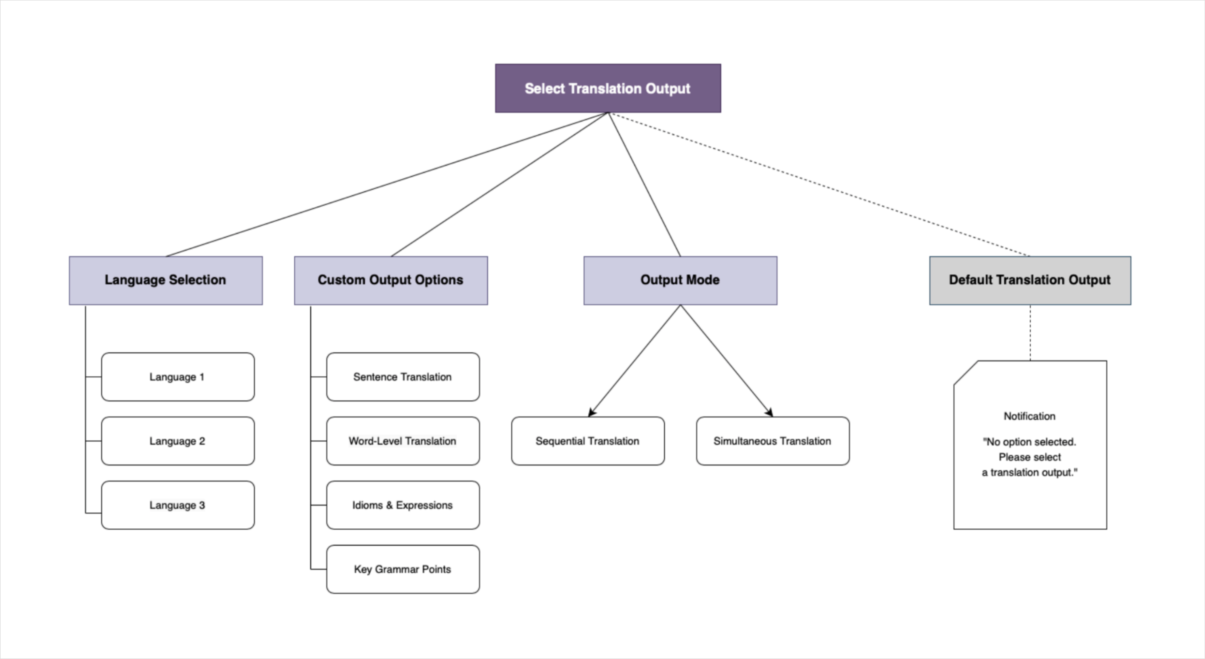


Figure 3. Examples of Options Available in General Translation

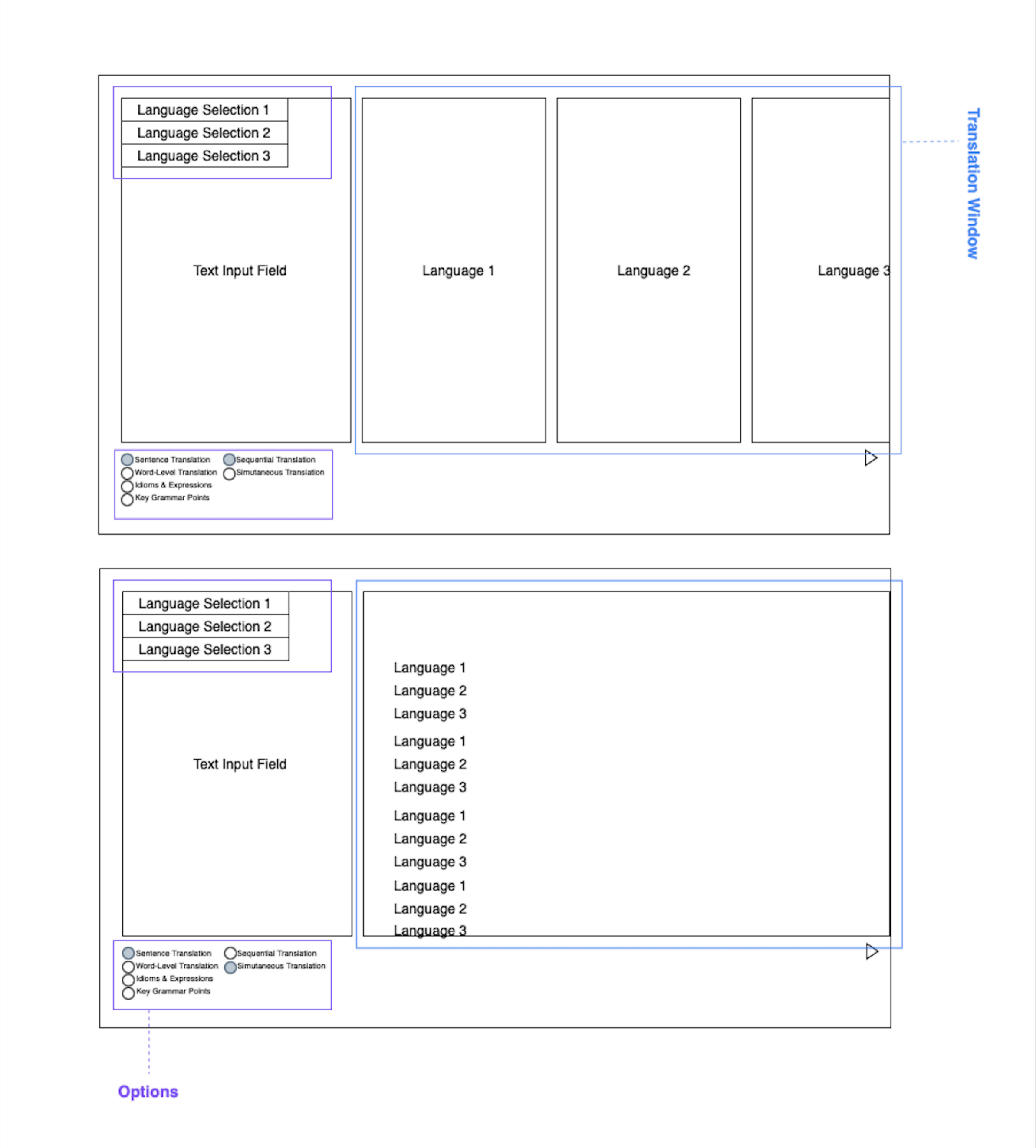


Figure 4. Initial Idea of Comparison of Sequential and Simultaneous Translation Windows

The second process focuses on **multilingual tense comparison**, a feature designed to facilitate cross-linguistic tense analysis. Users initiate this function by selecting the languages they wish to compare. The system then generates a comparative tense table, allowing for a systematic examination of tense structures across multiple languages. The results are displayed within the UI, with options for users to save or share the comparison data.

This application integrates both general translation and multilingual tense comparison functionalities, addressing diverse needs in language translation and linguistic analysis.

While the general translation workflow primarily focuses on lexical and syntactic processing, the multilingual tense comparison feature provides a structured framework for understanding temporal variations across languages. By combining these elements, the system serves as a comprehensive tool for both translation and cross-linguistic grammatical studies.

2.3 Django Directory Structure

Figure 5. Django MVT Architecture

The adoption of Django's project structure in my project aims to maximize modularity, scalability, and maintainability. Django follows the MVT (Model-View-Template) architecture, and effectively managing this architecture requires the separation of functionalities into distinct directories and files. This structured approach enhances code readability, enforces a clear separation of concerns, and optimizes development efficiency.

1. DjangoStructure/ - Project root directory
    a. project/ - Core Django settings and global URL management
        i. init.py - Marks directory as a Python package
        ii. settings.py - Project settings (DB, Middleware, Installed Apps, etc.)
        iii. urls.py - Global URL routing
        iv. asgi.py - Asynchronous server gateway interface
            Used for asynchronous server
            Needed for real-time functionality
        v. wsgi.py - Web server gateway interface
            Connects Django with web server
            Handles Django's default web pages
    b. translator/ - App directory (modular functionality)
        i. init.py - Marks directory as a Python package
        ii. views.py - Handles translation-related logic
        iii. urls.py - Defines URL patterns for the app
        iv. models.py - Database models
        v. serializers.py - Django REST Framework serializers
        vi. admin.py - Django Admin settings
        vii. apps.py - Django app configuration
        viii. migrations/ - Stores database migration files
    c. static/ - Static files (CSS, JS, images, etc.)
    d. templates/ - HTML templates

Figure 6. Example of Django Project Structure

For instance, in this structure, *views.py* is responsible for handling client requests, *models.py* manages database interactions, and *urls.py* defines the API endpoints utilized within the application, ensuring a clear separation of concerns. Moreover, *settings.py* enables centralized configuration management, allowing seamless adaptation to different deployment environments, while *migrations/* facilitates the secure tracking and application of database schema modifications. Adhering to this structured approach ensures that the project remains maintainable as it scales, preserving code consistency when integrating new functionalities or expanding existing ones. Furthermore, the use of *static/* and *templates/* directories enables the separation of static assets and HTML templates, thereby establishing a distinct boundary between frontend and backend responsibilities. Consequently, this structure is not merely a means of organizing files but a fundamental architectural strategy that leverages Django's strengths to enhance collaboration, maintainability, and scalability.

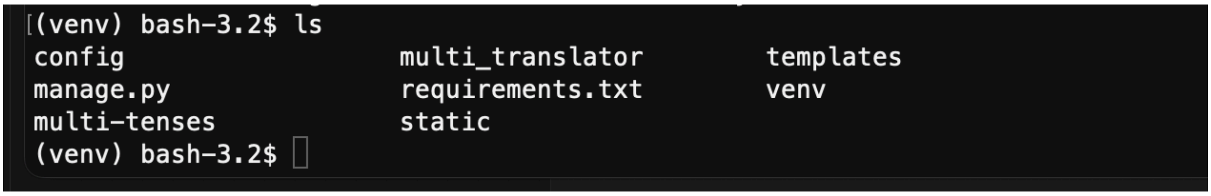
이부분 업데이트

웹 서버 연ㄱ결

어려움

서버 실행 오류

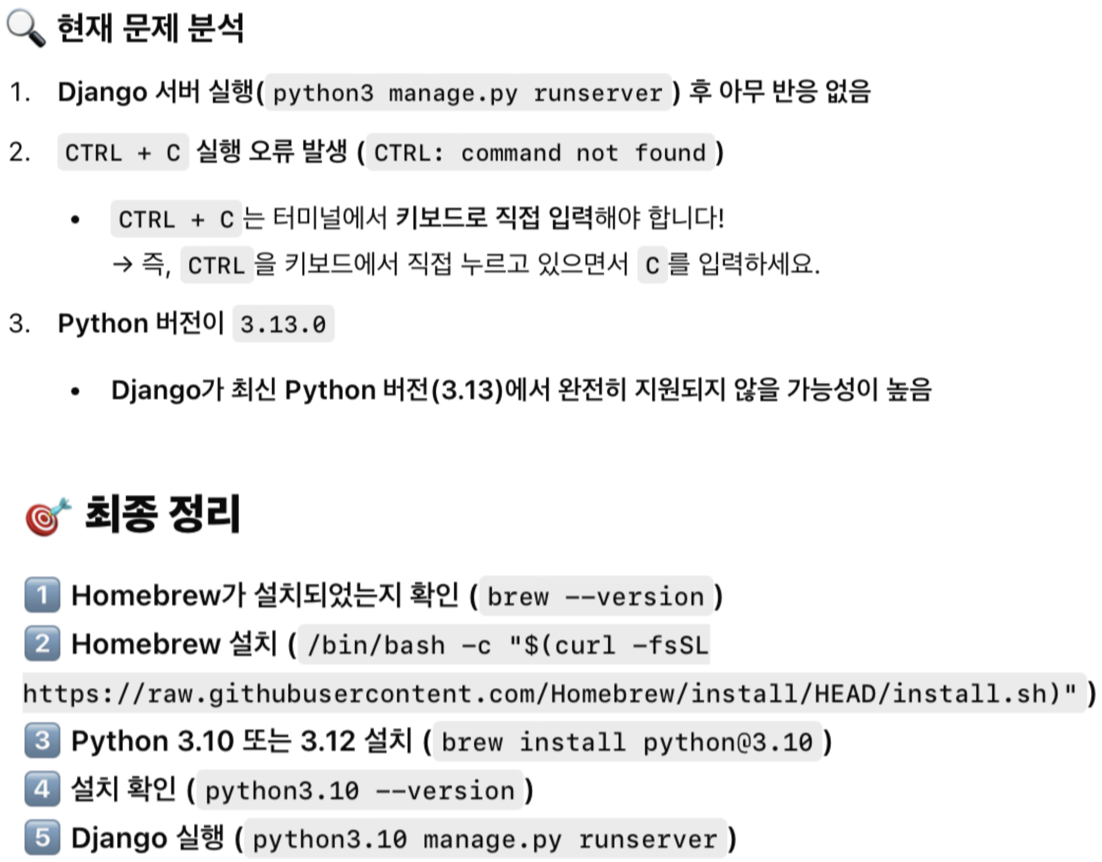
Django 서버 실행 오류



```
[(venv) bash-3.2$ ls
config                  multi_translator        templates
manage.py               requirements.txt        venv
multi-tenses            static
(venv) bash-3.2$ 
```

Django 서버를 실행가능-해야하지만 안됨.

Django 프로젝트가 올바른 구조로 되어 있음 ->



🔍 현재 문제 분석

1. Django 서버 실행(`python3 manage.py runserver`) 후 아무 반응 없음
2. `CTRL + C` 실행 오류 발생 (`CTRL: command not found`)
   - `CTRL + C`는 터미널에서 **키보드로 직접 입력**해야 합니다!
     → 즉, `CTRL`을 키보드에서 직접 누르고 있으면서 `C`를 입력하세요.
3. Python 버전이 `3.13.0`
   - Django가 최신 Python 버전(3.13)에서 완전히 지원되지 않을 가능성이 높음

🎯 최종 정리

1️⃣ Homebrew가 설치되었는지 확인 (`brew --version`)
2️⃣ Homebrew 설치 (`/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"`)
3️⃣ Python 3.10 또는 3.12 설치 (`brew install python@3.10`)
4️⃣ 설치 확인 (`python3.10 --version`)
5️⃣ Django 실행 (`python3.10 manage.py runserver`)

Python 3.13 설치되어있었지만 3.10/3.12에서만 가능.

3.2 Tentative Schedule

Week	Activities	Objectives
5	**System Architecture**	Designing Data Flow and System Architecture for a scalable web-based multilingual learning application.
6	**Prototype Architecture**	Creating initial wireframes and prototypes to define user flows and interface elements.
7	**Backend Development - Database & API**	Developing backend with Django for managing user data, translations, and language processing.
8	**Backend Development - API Implementation & Testing**	Integrating DeepL API and Google Cloud Translation API for real-time multilingual translations.
9	**Frontend Development - Basic UI Implementation** **Designing the Primary Translation Interface**	Developing core UI components using React and Next.js to create a responsive and dynamic user interface. Developing an intuitive translation dashboard with interactive elements for easy navigation and real-time results.
10	**Core Features - Translation & Word Analysis** **Real-time Multilingual Translation Testing**	Implementing word analysis features, including synonyms, antonyms, and example sentences powered by AI-based NLP models. Ensuring accurate, fast, and seamless language

		translation capabilities through rigorous testing.
11	**User Interactivity - Highlight & Instant Lookup**	Implementing a text selection feature for instant word definitions, grammar explanations, and synonym suggestions.
12	**Testing & Debugging - Core Features**	Debugging translation and word analysis functionalities for smooth performance and accuracy.
13	**UI/UX Improvements and Usability Testing**	Refining the user interface and experience based on usability tests and feedback.
14	**Finalizing & Optimization**	Performance Optimization: Enhancing speed with network request optimization and caching mechanisms. Finalizing UI Design: Ensuring consistency and accessibility in design elements.
15	**Feedback Collection & Revisions**	Collecting feedback and making necessary improvements.
16-17	**Final Testing** **Documentation & Future Scalability**	Ensuring system stability and conducting final tests. Final Code Refinement: Cleaning up the codebase and improving maintainability. Project Documentation: Preparing technical documentation to support future scalability.

---

4. Practical Applications

4.1 Use Cases

This application can be fundamentally utilized in three distinct ways: single-language translation, multi-language translation, and multi-tense table generation. Each function is designed to optimize linguistic accessibility and facilitate advanced multilingual text processing.

1) The single-language translation feature enables users to input text and select a target language for translation. For instance, if a user enters "Climate change is a pressing global issue." and selects French, the system generates *"Le changement climatique est un problème mondial urgent."*. Furthermore, users may access additional linguistic insights such as word-level translations (e.g., pressing → *pressant*, global → *mondial*) and grammatical analysis, highlighting syntactic structures and tense usage.

2) The multi-language translation function allows users to translate text into up to three different languages simultaneously, fostering comparative linguistic analysis. For example, if a user inputs "Technological advancements are reshaping the global economy." and selects French,

Spanish, and German, the system produces:

> *French: "Les avancées technologiques transforment l'économie mondiale."*
> *Spanish: "Los avances tecnológicos están transformando la economía global."*
> *German: "Technologische Fortschritte verändern die Weltwirtschaft."*

Users can further customize the translation process by choosing between two distinct modes: Simultaneous Translation, where each sentence is instantly translated into multiple languages, and Sequential Translation, where the entire text is translated into one language before being processed into the next. This structural flexibility accommodates diverse linguistic and academic needs.

3) The Multi-Tense Table Generation feature facilitates the comparative analysis of verb conjugations across multiple languages. When a user requests a tense comparison table for English, French, and Korean, the system generates a structured table that displays verb conjugations across various tenses. For example:

Tense	English	French	Korean
Present	*I analyze*	*J'analyse*	나는 분석한다
Past	*I analyzed*	*J'ai analysé*	나는 분석했다
Future	*I will analyze*	*J'analyserai*	나는 분석 할 것이다
Conditional	*I would analyze*	*J'analyserais*	나는 분석할 것이다 *(*가정법*)*

This feature is especially valuable for linguists, researchers, and language learners aiming to analyze cross-linguistic grammatical structures. Additional enhancements, such as irregular verb detection and contextual usage insights, further enhance the system's analytical capabilities.

By integrating these functionalities, this application serves not only as a translation tool but also as an advanced linguistic research platform, enabling precise multilingual communication and comprehensive grammatical analysis.

4.2 Examples of Potential Applications

1) AI-Enhanced Language Learning Tool for Multilingual Education

This application serves as an advanced tool for foreign language learners and multilingual users, setting itself apart from conventional translation software by offering a more comprehensive linguistic analysis. Rather than merely providing word-for-word translations, the system facilitates real-time language processing, enabling users to develop a deeper understanding of linguistic structures. Unlike standard translation tools, this system grants instant access to word definitions, grammatical explanations, idiomatic expressions, synonyms, and antonyms, eliminating the need for manual dictionary searches and streamlining the learning process.

For instance, when a user inputs an excerpt from a news article, the system not only translates the text into the selected target language(s) but also delivers contextual vocabulary explanations, idiomatic expressions, grammatical structures, and example sentences. Additionally, by providing synonyms and antonyms, the system allows learners to expand their lexical and expressive range more efficiently. This integrated approach to language acquisition transforms passive translation into an interactive, intuitive learning experience, enabling users to master multiple languages with greater efficiency and precision.

2) Integration with Email Communication

Integrating this system into email platforms would significantly enhance the efficiency and accuracy of composing and interpreting foreign-language correspondence. When receiving an email in a foreign language, users could simply highlight unfamiliar words or phrases, prompting the system to provide not only translations but also synonyms, antonyms, and grammatical insights in real time.

Similarly, while composing emails, users seeking more natural phrasing or precise terminology could utilize the system’s real-time translation suggestions and grammar refinement tools, ensuring that their writing is both linguistically accurate and contextually appropriate. By embedding this system within email clients, language barriers can be reduced, facilitating seamless multilingual communication across academic, professional, and global contexts.

3) AI-Powered Language Learning and Translation Model Development

The advancement of AI-driven language learning and translation technologies is crucial in enhancing the contextual accuracy and linguistic sophistication of machine translation systems. While AI-based translation models such as ChatGPT have demonstrated significant progress, they still struggle with contextual nuance, cultural specificity, and the accurate transmission of meaning across languages.

By leveraging this application, researchers can systematically compare real-time translations across multiple languages, detect grammatical inconsistencies, and refine natural expressions through large-scale linguistic data collection. This iterative process enhances AI training methodologies, enabling translation models to achieve higher precision, greater contextual adaptation, and more human-like linguistic output. As a result, this system contributes directly to the development of next-generation AI translation technologies, fostering a future where machine translation approaches near-human proficiency in both accuracy and contextual awareness.

Reference

Django MVT

https://medium.com/@CodeMaple/understanding-django-mvt-architecture-and-view-functions-django-full-course-for-beginners-lesson-39c8da093b44