AI Translation Project Report

Project Manager and Technical Builder: **Emile Zounon**

Section 1: Technical Recommendations from the Builder

1. Overview of the tools

Throughout this project, Emile tested three AI translation tools, each with distinct capabilities, integration potential, and cost implications:

Tool	Model / Version	Access Method	Primary Role in Project	Cost & Usage
ChatGPT	GPT-4 (June 2024 snapshot)	OpenAI API (personal account)	Rapid, flexible translation via scripting and Excel API integration	**$20/month** (GPT Plus subscription with **43,027 tokens** used)
DeepL	DeepL Pro (latest as of June 2025)	DeepL API(perosna	High-accuracy, batch translation for formal/instructional text	**Free** for 500,000 characters. **188,632 characters used** (38% of limit)
Watsonx	Granite-3B-Instruct (8B model)	Manual interface via watsonx.ai	Used manually due to API failure; served as slower fallback	**Free** (internal platform access. Used all the free 300,000 tokens)

2. Method of Translation: Two Approaches Tested

Manual Copy-Paste Method (Watsonx only)

Initially, Emile attempted to automate Watsonx using the same API-based script used for the other tools. However, he encountered persistent issues: either the full text wasn't being translated, or the script returned errors. Due to time constraints and a lack of documentation, he was unable to resolve the integration.

To troubleshoot, Emile contacted a Watsonx junior developer who offered suggestions to improve the code. Unfortunately, they were unavailable to support full debugging. As a result, he completed all Watsonx translations manually: copying and pasting each sentence into the tool and then pasting the results back into the spreadsheet.

- **Time Required:** ~6 hours total (45 minutes per language × 8 languages)
- **Challenges:** Manual repetition, higher risk of error, and poor scalability
- **Status:** Currently not viable for scaled or repeatable workflows

API Scripting Method (Used for GPT and DeepL)

For GPT and DeepL, Emile wrote scripts that connected Excel directly to each platform's API. This allowed him to batch-translate all course content directly in the spreadsheet.

- **Setup Time:** 1 hour to write and test the script
- **Execution Time:** 30 minutes for complete translation in 9 languages
- **Advantages:** Minimal formatting issues, fast, and fully replicable
- **Scalability:** High-ready for larger projects with minimal adjustments

3. Builder's Recommendation

From Emile's perspective, **API-based scripting** is the most efficient, scalable, and production-ready method for multilingual translation. It dramatically reduces human workload and enables consistent outputs across languages and tools.

Method	Time Investment	Scalability	Tech Effort	Cost	Recommendation
Manual Copy & Paste	6 hours	Low	None	Free (Watsonx.ai)	Not recommended
API Scripting (GPT)	0.5 hours total	High	Moderate	$20/month (43,027 tokens)	Recommended
API Scripting (DeepL)	0.5 hours total	High	Moderate	Free (188,632 of 500,000 max characters)	Recommended

Next Step for Watsonx: With more time or assistance from Watsonx engineering, the API issues could likely be resolved and brought into parity with GPT and DeepL. Until then, its use should be limited to smaller, low-priority tasks.