

Project Proposal Reflection

🧡 Team Scrum-ptious 🧡

Intended use of the system: who and how will use the system (20 pts)

The system is designed to ensure content parity across different language versions of Wikipedia articles and, in the future, other wiki-based platforms. The goal is to provide accurate, consistent, and culturally appropriate information across the language, especially for underrepresented languages. This guarantees equal access to knowledge globally.

- **Wikipedia contributors:** Identify missing, outdated, or inaccurate information by comparing different language versions. AI-generated suggestions help editors update and improve articles, ensuring content parity.
- **Researchers and Academics:** Analyze translation accuracy, information bias, and linguistic disparities using the semantic comparison tools. The system helps measure translation alignment and identify multilingual content quality patterns.
- **Educational & Inclusivity-Focused Organizations:** Improve digital accessibility and multilingual education by ensuring underrepresented languages receive accurate and comprehensive content, promoting equal access globally.
- **Other Wiki-Based Platforms & Content Managers:** While the system is designed for Wikipedia, it can be adapted for any MediaWiki-based platform or similar collaborative knowledge bases. Scraping and comparison tools help maintain content consistency and translation accuracy across multilingual sites.

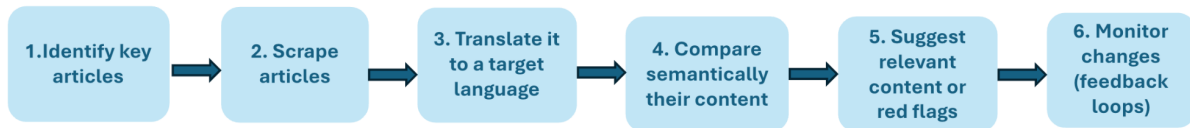
Its overall functionality: what will the system do, how will the system help its users accomplish their tasks. If any parts of the system already exist, specifically focus on the functionality that your team will be adding to the system. (15 pts)

The system will provide machine translations into the target language. The target language page will be compared to translated pages to locate missing, inaccurate, or supplemental information. Editors can use the provided translations and suggested changes to much more rapidly improve article quality.

Goals:

- **Translational accuracy:** The system will use advanced machine learning models like T5 and Marian to translate articles from one language to another.
- **Semantic comparison:** The system will compare the translated text with the existing version of the article in the target language, locating similarities. BLEU and other algorithms will be used to compare meaning rather than exact phrasing.
- **Actionable suggestions:** The system will suggest changes to bring the articles into informational parity. An editor can review and incorporate the suggestions with minimal work.

Roadmap



Our role: As the only component we are confident is near completed is the method for scraping Wikipedia data, our team will focus on the following areas:

- Front end application UI/UX: Implement the user interface based on Figma designs.
- Middleware/APIs: Develop middleware and APIs to connect the user interface and AI processing systems.
- AI translation: leverage pre-trained AI models to supply translations that are not just accurate to the words being translated, but also the idioms of the language.
- Comparison metrics: Research various comparison techniques, from algorithms to AI, to determine a best-fit for the project's initial needs.

Main components of the system: break down the system into logical or architectural components and provide the rationale for this breakdown (15 pts)

- Data acquisition module: Pull content from Wikipedia in a variety of languages. Scrape pages' content and references to ensure that data is up-to-date. Because this is a separate module, it becomes a much smaller task to supply a different data source for comparisons.
- Translation engine: Use pre-trained LLMs like T5 and Marian to translate content from one language to another. Supporting a wide variety of translation techniques and models allows editors to select the translation model that suits their language and needs best.
- Semantic analysis: Compare translations with the existing article and check for missing sections, biases, or other inconsistencies. The goal of separating out this component is to make it easier to interchange comparison algorithms.
- Suggestion engine: Based on the results of semantic analysis, this component will generate article modification recommendations, telling editors exactly where changes are needed. This consists of determining where information may be missing or what sections need to be modified.
- User Interface: Display translations, semantic comparisons, and suggestions in an easy-to-understand format. Visual cues like color coding will help users identify problem areas.