

Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	31 January 2025
Team ID	LTVIP2026TMIDS66199
Project Name	TransLingua: AI-Powered Multi-Language Translator
Maximum Marks	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Text Translation	<ul style="list-style-type: none">• User can enter text to be translated.• User can select source language.• User can select target language.• System generates translated text using AI.
FR-2	Input Validation	<ul style="list-style-type: none">• Validate that text field is not empty.• Ensure source language is selected.• Ensure target language is selected.• Prevent translation if source and target languages are the same (optional validation).• Display appropriate error message for invalid input.
FR-3	AI Model Integration	<ul style="list-style-type: none">• Create structured translation prompt using user input.• Send prompt to Gemini Generative AI model via API.• Receive translated response from AI.• Extract and format the AI-generated text properly.
FR-4	Output Display	<ul style="list-style-type: none">• Display translated text in a clear and readable format.• Allow user to copy translated text.• Show user-friendly error message if API fails.
FR-5	Multi-Language Support	<ul style="list-style-type: none">• Provide multiple language options in dropdown menu.• Support translation between various language pairs.
FR-6	Error Handling & Performance	<ul style="list-style-type: none">• Handle API timeout or runtime errors gracefully.• Ensure translation is generated within acceptable response time.

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The application should provide a simple, intuitive, and user-friendly interface using Streamlit.
NFR-2	Security	The API key must be securely configured and not exposed publicly. User input should not be stored permanently.
NFR-3	Reliability	The system should handle invalid inputs and API errors gracefully without crashing.
NFR-4	Performance	The application should generate travel itineraries within a reasonable response time.
NFR-5	Availability	The system should be accessible whenever the Streamlit server is running and internet connectivity is available.
NFR-6	Scalability	The system architecture should allow future enhancements such as budget-based customization and additional travel features.