2. REQUIREMENT ANALYSIS

2.1 Introduction

Requirement analysis is a critical phase in software development. It defines system functionality, constraints, and user expectations.

For TransLingua, requirements were gathered based on use cases in business, education, and travel domains.

2.2 Functional Requirements

Functional requirements describe what the system should do.

FR1: User Input

- The system must allow users to enter text for translation.

FR2: Language Selection

- The system must provide dropdown options for selecting:
    - Source language
    - Target language

FR3: Translation Processing

- The system must send the input text and selected languages to the Gemini Pro model.

FR4: Output Display

- The system must display the translated output clearly.

FR5: Error Handling

- The system must handle API errors gracefully.

2.3 Non-Functional Requirements

These define performance and quality attributes.

Performance

- Translation must be generated within a few seconds.

Reliability

- The system should provide consistent results.

Security

- API keys must be stored securely using .env file.

Usability

- Interface must be simple and user-friendly.

Scalability

- System should support future feature extensions.

2.4 Hardware Requirements

- Processor: Intel i3 or above
- RAM: Minimum 4GB
- Internet Connection
- Laptop/Desktop

2.5 Software Requirements

- Python 3.9+
- Streamlit
- google-generativeai library
- python-dotenv
- VS Code / Any IDE
- Web Browser

2.6 System Constraints

- Internet dependency
- API rate limits
- Model availability
- Language support limitations

2.7 Use Case Analysis

Use Case 1: Business Translation

Actor: Business Manager
Action: Translate marketing content
Outcome: Multilingual business communication

Use Case 2: Academic Research

Actor: Researcher
Action: Translate research paper
Outcome: Cross-border collaboration

Use Case 3: Travel Assistance

Actor: Traveler
Action: Translate menu/sign board
Outcome: Easy communication in foreign country