

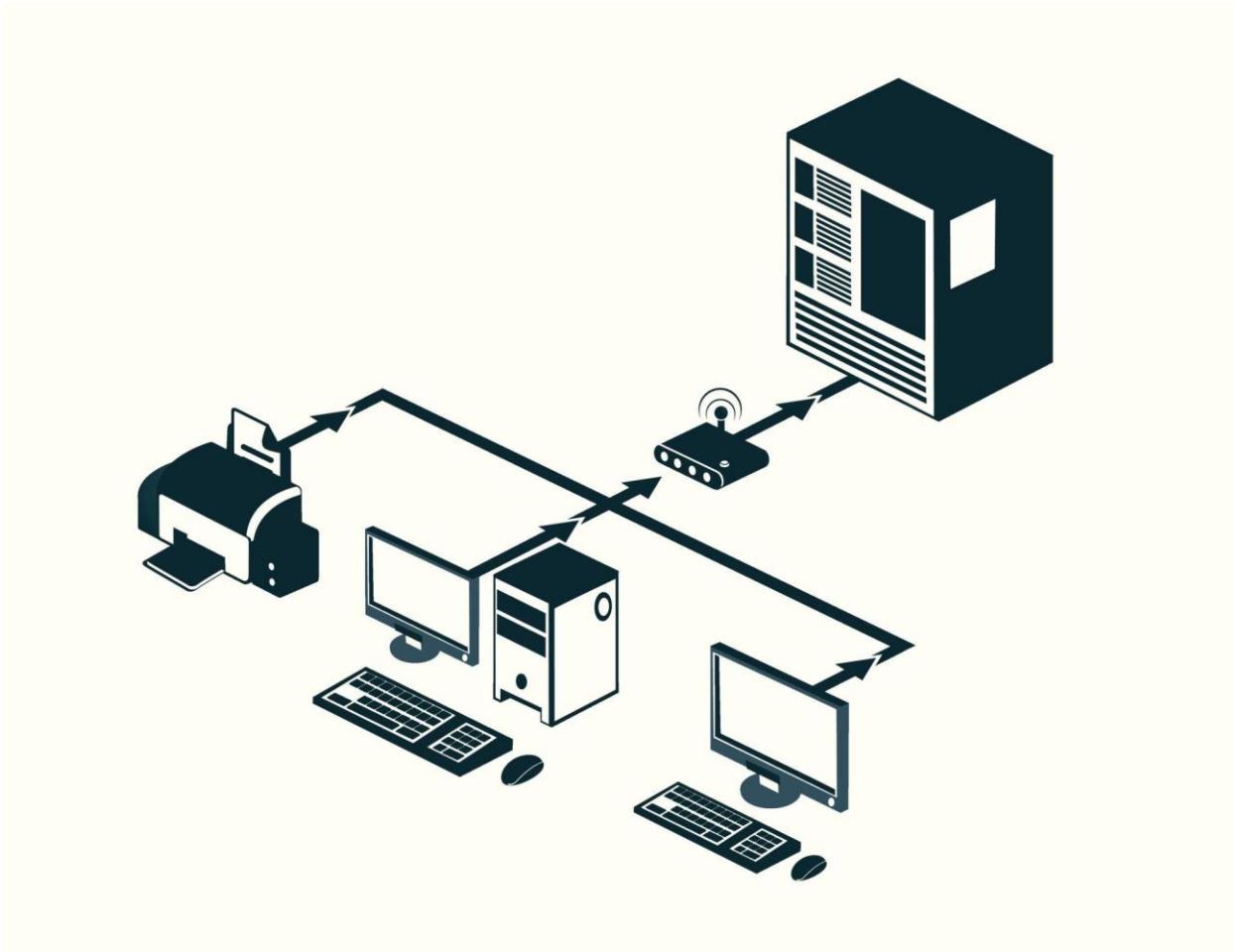
3. PROJECT DESIGN PHASE

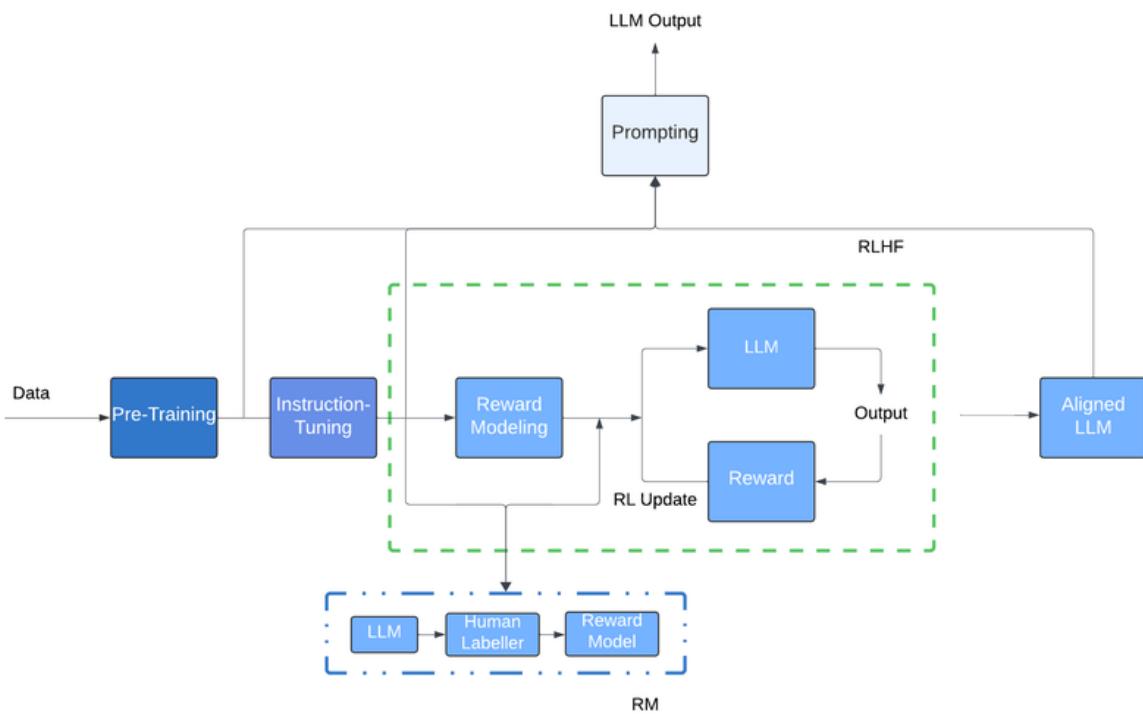
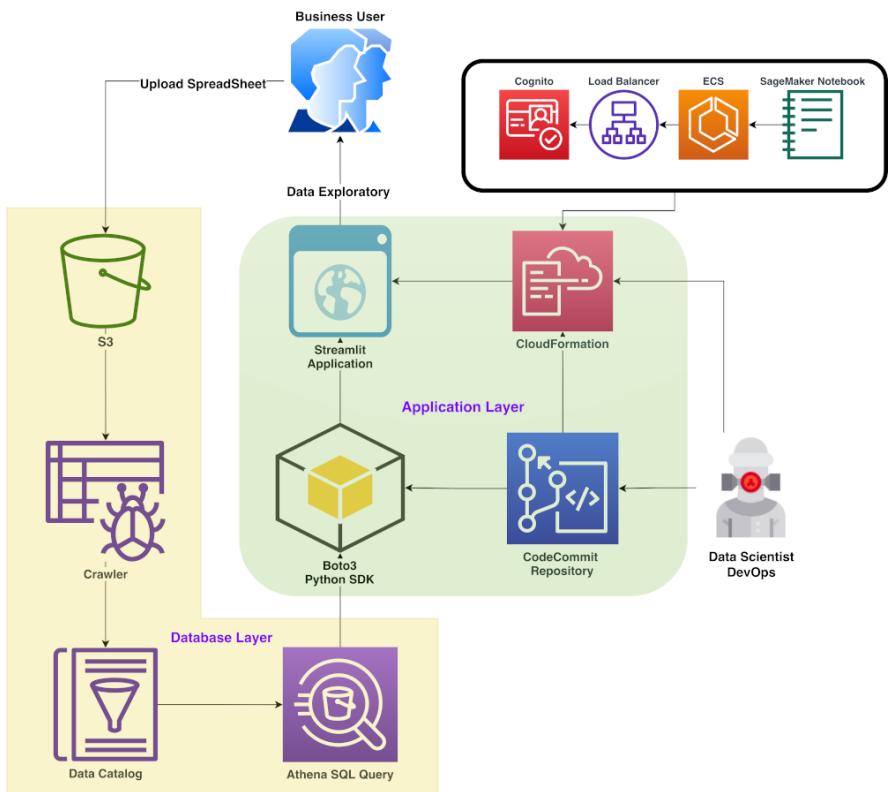
3.1 Introduction

The Project Design Phase defines the system structure, architecture, workflow, and component interaction. This phase ensures that the system is scalable, modular, and easy to maintain.

For TransLingua, the design follows a simple **Client–Server Architecture** integrated with a Generative AI model.

3.2 System Architecture Design





Architecture Explanation

The system consists of:

1. Frontend (Streamlit UI)

- Accepts user input
- Language selection
- Displays translated output

2. Backend (Python + Gemini API)

- Constructs dynamic prompts
- Sends request to Gemini model
- Receives AI-generated translation

3. AI Model (Gemini 1.5 Flash)

- Processes language translation
 - Generates context-aware output
-

3.3 Data Flow Design

Step-by-Step Flow:

1. User enters text.
 2. User selects source language.
 3. User selects target language.
 4. Prompt is dynamically generated.
 5. Request is sent to Gemini API.
 6. Model generates translation.
 7. Response is displayed in UI.
-

3.4 Module Design

The system is divided into modules:

Module 1: User Interface Module

- Streamlit components
- Input text area
- Language dropdown
- Translate button

Module 2: Translation Logic Module

- Prompt construction
- Model initialization
- API interaction

Module 3: Configuration Module

- API key management
 - Environment setup
-

3.5 Database Design

Currently:

- ✗ No database required
- Stateless application
- Real-time processing

Future enhancement:

- Store translation history
- User login system
- Save translated documents