|  |
| --- |
| **LinguoFix**  Project Charge Document (M101) |

1. **Project Information**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Name:** | LinguoFix | **PlanView Number:** | 1 |
| **Project Sponsor:** | Nazar Melnyk | **Project Size:** | Medium |
| **Project Director:** | Vitalii Dorosh | **Project Start Date:** | 03/03/2025 |
| **Project Manager:** | Mykola Telka | **Project End Date:** | 05/05/2025 |

1. **Project Leads**

| **Lead Name** | **Division** | **Unit** |
| --- | --- | --- |
| Mykola Telka | Software Development | Project Manager & Team Lead |
| Svitlana Kontsohrada | Design | UX/UI Designer |
| Dmytro Kaliukh | Development | Backend Developer |
| Huska Andrii | Development | Front-end Developer |
| Diana Mudrak | Quality Assurance | QA Specialist |

# Project Statement

| **Project Area** | **Description** |
| --- | --- |
| Background: | LinguoFix is an AI-powered web application designed to improve written communication by automatically detecting and correcting grammar, spelling, and stylistic errors. It also provides real-time translations for multiple languages and their variations, serving students, professionals, and non-native speakers. |
| Project Vision Statement: | To provide a reliable, AI-driven solution that enhances writing quality and promotes effective communication globally through automatic text correction and multilingual support. |
| Objective: | To develop and deploy an AI-powered platform that offers accurate grammar, spelling, and style correction, alongside multilingual translation, with a user-friendly web interface. |
| Scope: |  |
| Included in scope: | Development of AI-based grammar checker, multilingual translation module, user-friendly web interface, API integrations, and optional history tracking for users. |
| Excluded from scope: | Offline functionality, advanced text analysis (e.g., plagiarism detection), mobile applications (web-based only in initial release). |
| Impacts: (Organizational & Technical) | Organizational Impact  Improves productivity for businesses, academics, and professionals by ensuring high-quality written content. Reduces the need for external proofreading and translation services.  Technical Impact  The project will enhance AI capabilities in text correction and translation, leveraging external APIs for language processing. It will also ensure a smooth integration between the AI system and the web platform. |
| Dependencies: | Reliance on external language processing APIs (e.g., Grammarly, DeepL) for translation and correction accuracy. Integration with backend infrastructure for real-time functionalities. |
| Assumptions & Constraints: | Assumptions  AI models will provide sufficiently accurate corrections for production use. Users will have stable internet connections for real-time corrections.  Constraints  Limited budget for premium API usage, compliance with GDPR and data privacy regulations. |

# 

# High-level Requirements

| **Requirement Area** | **Projected Steps/Comments** |
| --- | --- |
| AI-Powered Grammar and Spell Checking | Implement a real-time grammar and spell checking system using AI-driven models. This includes integrating APIs such as Grammarly or developing a custom solution. |
| Multilingual Translation | Integrate a multilingual translation module supporting various languages and language variations (e.g., British and American English). Use APIs like DeepL. |
| User Interface (UI) | Design a responsive, user-friendly web interface that provides easy text input, correction suggestions, and translation options. Ensure cross-device accessibility. |
| API Integration | Integrate external language processing APIs (e.g., Grammarly for grammar checks, DeepL for translations) for enhanced text correction and translation accuracy. |
| User History Feature | Develop an optional feature that allows users to save and review their previous text corrections, enhancing user experience and usability. |
| Technical Environment | Develop the system using modern web technologies (e.g., React for frontend, Node.js for backend). Ensure scalability, performance optimization, and secure API handling. |

# High-level Deliverables

| **Deliverable Type** | **Description** |
| --- | --- |
| Business Process | AI-Powered Grammar Checker: A fully functional tool that identifies and corrects grammatical, spelling, and stylistic errors. |
| Business Process | Multilingual Translation Module: A system providing real-time translations for multiple languages, including language variations. |
| Business Process | Web-Based User Interface: A responsive, user-friendly UI that allows users to interact with the grammar checker and translation module. |
| Business Process | User History Feature: An optional feature that enables users to save and review their previous text corrections. |
| Project Management | Project Plan and Timeline: A detailed project plan outlining key milestones, timelines, and resource allocation for the project's success. |
| Project Management | Risk Management Plan: A document outlining potential risks and corresponding mitigation strategies for the successful delivery of the project. |
| Project Management | Testing and Optimization Report: A report that tracks the results of testing phases, including bug fixes, optimizations, and performance improvements. |
| Project Management | Deployment and User Feedback Report: A document detailing the deployment phase and feedback from initial users for further improvement. |

# High-level Timeline

| **Milestone** | **Target Date** | **Date Achieved** |
| --- | --- | --- |
| Requirements Gathering and Planning | 2 weeks |  |
| Development | 4 weeks |  |
| Testing and Optimization | 2 weeks |  |
| Deployment and Initial User Testing | 2 weeks |  |

# High-level LOE Estimates

| **Description of Effort** | **Estimated # Hours across # Days to Complete** | **Comments** |
| --- | --- | --- |
| Requirements Gathering and Planning | 40 hours (5 days) | Includes meetings, documentation, and planning activities. |
| Development (Backend and Frontend) | 160 hours (20 days) | Involves coding the core features and user interface. |
| UI/UX Design | 60 hours (7.5 days) | Design the user interface and ensure ease of use. |
| API Integration | 40 hours (5 days) | Integrate external APIs for grammar and translation. |
| Testing and Optimization | 60 hours (7.5 days) | Includes testing, debugging, and performance improvements. |
| Deployment and Initial User Testing | 40 hours (5 days) | Prepare for deployment and initial testing by users. |

# Project Team

| **Project Groups** | **Division/Units Represented** |
| --- | --- |
| Core Group(s): | Development Team (Backend, Frontend, UI/UX) |
| Subject Matter/Expert(s): | Language Processing, AI Algorithms |
| Technical Expert(s): | API Integration, System Architecture |
| Other(s) (please describe): | Quality Assurance, Testing |

# 

| **Steering Committee Member** | **Division/Units Represented** |
| --- | --- |
| Mykola Telka | Project Manager & Team Lead |
| Svitlana Kontsohrada | Designer (UI/UX) |
| Dmytro Kaliukh | Backend Developer |
| Diana Mudrak | QA Tester |
| Huska Andrii | Front-end Developer |

# Project Status Reports

|  |  |
| --- | --- |
| **Status Report Owner:** | Mykola Telka |

|  |  |  |
| --- | --- | --- |
| **Status Report Audience:** | Project Sponsor | Stakeholders |
|  | Steering Committee | Core Development Team |

*Internal Use Only*

**Document Change Control**

**Project Charge Document (M101)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version Number** | **Date** | **Revision Author** | **Description** |
| 1.0 | 2025-03-02 | Mykola Telka | Initial draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

*This document is owned and maintained by MAIS Shared Services, Project Support. Any changes to template format and content must adhere to departmental document management standards.*