**Project Name: P3 Next Word Predictor**

**Surender Reddy Erigela (100941347)**

**Rohith Gundlapally (100941348)**

**Sai Siddharth Mothe (100947571)**

**Rakesh Babu Konakanchi (100940386)**

**Preliminary Scope**

The P3 - Next Word Predictor project aims to develop a cloud-hosted application that utilizes both software (SW) and an AI agent for next-word prediction. The project's preliminary scope includes:

1. Software Development (SW)

* Text Tokenization: Break down input text into tokens or words for analysis.
* Word Frequency Analysis: Conduct basic statistical analysis to identify the most common words following certain sequences.
* Simple Predictive Model: Implement a basic model like N-gram to predict the next word based on the previous one or two words.
* Database of Common Phrases: Create a lookup system for common phrases or collocations to enhance predictions.
* User Interface: Develop a user-friendly interface for text input and display predictions.

2. AI Agent

* Deep Learning Models: Implement advanced AI models such as RNN (Recurrent Neural Networks), LSTM (Long Short-Term Memory), or Transformer models to understand context better and provide more accurate predictions.
* Contextual Understanding: Utilize AI models like BERT (Bidirectional Encoder Representations from Transformers) or similar technologies to analyse deep contextual word relationships.
* Personalization: Learn from individual user typing patterns and preferences to personalize predictions.
* Multi-language Support: Enable the system to predict words accurately in multiple languages.
* Handling Typos and Grammar: Integrate a system to understand and correct typos and grammatical errors before making predictions.
* Predictive Text Expansion: Suggest not only the next word but also possible continuations of the sentence.
* Voice Input Compatibility: Allow predictions based on both voice input and text.
* Adaptive Learning: Continuously improve the AI model based on user feedback and new data.
* Real-Time Learning: Update the predictive model in real-time as more text is typed or corrected by the user.
* API Integration: Develop APIs for easy integration with other applications and services.

**Technology Stack**

The following technology stack will be used to implement the project:

* Cloud Hosting: AWS, Google Cloud, Azure
* Programming Languages: Python, JavaScript
* AI/ML Frameworks: [TensorFlow, PyTorch
* Database: PostgreSQL, MongoDB

**Use Cases for Project Evaluation and Tests**

During the project evaluation and testing phases, the following use cases will be followed by users to assess the functionality and performance of the Next Word Predictor:

1. **Basic Prediction**:
   * Users enter text, and the system predicts the next word based on the input.
2. **Personalization**:
   * Users provide input to personalize predictions, and the system adapts to their typing patterns.
3. **Multi-language Support**:
   * Users switch between different languages, and the system accurately predicts words in the selected language.
4. **Typo and Grammar Handling**:
   * Users intentionally introduce typos and grammatical errors, and the system corrects and predicts the intended words.
5. **Predictive Text Expansion**:
   * Users input a partial sentence, and the system suggests possible continuations for a coherent sentence.
6. **Voice Input Compatibility**:
   * Users use voice input, and the system accurately predicts the next word based on spoken words.
7. **Adaptive Learning**:
   * Users provide feedback on predictions, and the system learns and improves over time.
8. **Real-Time Learning**:
   * The system updates its predictive model in real-time as users type or make corrections.
9. **API Integration**:
   * Developers integrate the predictive model into third-party applications using APIs.