**Project Title:**
Worder - An AI-Powered Word Game

**Topic Selection:**
The project combines AI and NLP (Natural Language Processing) with game development to create an entertaining word game. It is relevant to computer science due to its integration of NLP and AI technologies into the gaming domain. We want to explore this domain as the influence of AI can be seen on various gaming industries.

## Introduction:
Worder is an AI-driven word game that leverages the power of AI and NLP to provide an engaging and challenging gaming experience. This project aims to develop a desktop application initially using Unity and later deploy it to the Play Store and App Store for mobile platforms. The game's primary objective is to showcase the potential of AI and NLP in the realm of entertainment, making it not only fun but also intellectually stimulating.

## Objectives:
- Develop a desktop version of Worder using Unity, focusing on cross-platform compatibility.
- Implement AI and NLP algorithms, specifically word2vec, to enhance the gameplay byproviding intelligent word similarity feedback.
- Create a two-player word-guessing game where players input words, and AI providesreal-time feedback on word similarity, making the game challenging and educational.
- Ensure a seamless user experience through a polished and intuitive user interface.
- Deploy the game on desktop platforms (Windows, macOS, Linux) and later on mobile platforms (iOS and Android).
- Demonstrate the capabilities of AI and NLP in a gaming context, highlighting the potential for educational gaming.

## Methodology:
- **Unity Development**:
  Utilize Unity for game development, including UI/UX design, game mechanics, and cross-platform compatibility. Figma will be used to prototype the layout.
- **AI Integration**:
  Integrate word2vec for AI-driven word similarity comparisons, enabling the game to analyze and provide feedback on player inputs.
- **GitHub Collaboration**:
  Collaborate using GitHub for version control, project management and efficient handling of code contributions.
- **Testing and Quality Assurance**:
  Conduct thorough testing and debugging to ensurethe game functions flawlessly on all target platforms. Unit tests will be performed regularly and the integration tests will be performed when the AI features are implemented.

## Expected Outcomes:
- A fully functional desktop version of Worder with AI-powered word similarity feedback, enhancing the gameplay experience.
- A polished user interface with a visually appealing design that enhances user engagement.
- Successful deployment of the game on desktop platforms (Windows, macOS, Linux).
- Deployment on mobile platforms (iOS and Android) for wider accessibility and user reach.
- A showcase of how AI and NLP can be seamlessly integrated into entertainment.

**Timeline:**
We will be meeting twice every week to discuss our project goals and accomplishments for improving the code. We will be virtual most of the time. The tentative tasks

**Weeks 1-2**: Conceptualization and Detailed Planning
**Weeks 3-6**: Unity Development, including UI/UX Design and Game Mechanics
**Weeks 7-9**: Integration of word2vec for AI-driven word similarity feedback
**Weeks 10-12**: Rigorous Testing and Debugging
**Weeks 13-16**: Deployment on Desktop Platforms (Windows, macOS, Linux) and Initial Mobile Development

**Group Work:**

**Prabhash GC**: Leads version control, testing, and UI/UX development. Responsible for ensuring a smooth user experience.
**Rohan Upadhyay**: Drives the core game mechanics development and ensures the efficient handling of pull requests on GitHub, focusing on gameplay functionality.

*This project aims to not only create an entertaining word game but also demonstrate the potential of AI and NLP in education and gaming. It will provide players with a challenging and intellectually stimulating experience while showcasing the capabilities of AI-driven entertainment.*