**Résumé des commandes**

Voici un résumé des commandes à utiliser pour chaque modification :

bash

Copier le code

git status # Vérifie les modifications

git add . # Ajoute toutes les modifications

git commit -m "Message de commit" # Enregistre les modifications avec un message

git push origin master # Envoie les modifications sur GitHub

let validWords = [];

// Charger la liste des mots depuis words.json

fetch('words.json')

    .then(response => response.json())

    .then(data => {

        validWords = data.words;

    })

    .catch(error => console.error('Error loading words:', error));

function submitWord() {

    const word = currentWordEl.value.toLowerCase().trim();

    // Vérification de la validité du mot

    if (isValidWord(word)) {

        usedWords.add(word);

        let points = word.length;

        if (word.includes(bonusLetter)) points += 5;  // Points bonus pour la lettre bonus

        score += points;

        updateScore();

        fetchDefinition(word, points); // Obtenir la définition et afficher les points

    } else {

        messageEl.textContent = usedWords.has(word) ? 'Word already used!' : 'Invalid word!';

        messageEl.style.color = 'red';

        score -= 1;  // Pénalité pour mot invalide ou déjà utilisé

        updateScore();

    }

    currentWordEl.value = '';

// Obtenir la définition d'un mot via l'API

function fetchDefinition(word, points) {

    const apiUrl = `https://api.dictionaryapi.dev/api/v2/entries/en/${word}`;

    fetch(apiUrl)

        .then(response => response.json())

        .then(data => {

            const definition = data[0]?.meanings[0]?.definitions[0]?.definition || "No definition found.";

            messageEl.textContent = `Correct! +${points} points. Definition: ${definition}`;

            messageEl.style.color = 'green';

        })

        .catch(error => {

            console.error('Error fetching definition:', error);

            messageEl.textContent = `Correct! +${points} points. Definition unavailable.`;

            messageEl.style.color = 'green';

        });

}