A Project Report

on

Spelling Practice Application With Feedback



by

Muhammad Bin Khalid BCPE 243046

Ahmad Arshad BCY 243001

A Project Report submitted to the

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

in partial fulfillment of the requirements for the degree of

BACHELORS OF SCIENCE IN ELECTRICAL ENGINEERING

Faculty of Engineering

Capital University of Science & Technology,

Islamabad

January, 2025

All rights reserved. Reproduction in whole or in part in any form requires the prior written permission of Muhammad Zubair Ali BEE183080, Mustafa Asad BEE183080, M Rehan BEE183023 or designated representative

[Table](#_Toc112428893) of Content

[1 Intoduction 2](#_Toc112428894)

[2 Problem 3](#_Toc112428895)

[3 Methodologies Used 4](#_Toc112428896)

[4 Code 5](#_Toc112428897)

[5 Results 6](#_Toc112428898)

[6 References 7](#_Toc112428900)

**Introduction**

The Spelling Practice Application is a console-based project written in C++ to improve spelling skills. It provides interactive features such as practicing spelling, adding new words, viewing a list of practice words, and deleting words. The program emphasizes user-friendly design with clear menus, input validation, and color-coded text for better readability.

The primary objective of this project is to create a practical tool that helps users enhance their spelling abilities while showcasing advanced C++ programming concepts such as file handling, vector manipulation, and ANSI escape codes for text formatting.

This application is especially useful for students, educators, and anyone who wants to work on their spelling in an engaging and structured manner.

**Problem**

The application addresses the following challenges:

* Lack of engaging tools for improving spelling skills.
* Difficulty in maintaining personalized spelling word lists for practice.
* Absence of immediate feedback for incorrect spellings.

By creating this project, we aim to provide a simple, interactive, and user-focused solution for these issues.

**Methodologies Used**

The following methodologies were applied during the development process:

File Handling: Used for storing and loading spelling words persistently.

Console User Interface (CLI): Designed using cout and ANSI escape codes for improved user interaction.

Input Validation: To ensure users input correct data types and avoid crashes.

Vector Manipulation: To manage and modify the word list dynamically.

ANSI Escape Codes: To introduce color-coded text for better readability.

Code Modularization: Divided tasks into smaller, reusable functions for clarity and maintainability.

**Code**

The application is structured with modular functions for better readability and reusability.

Core Functions:

displayMenu(): Displays the interactive menu.

getUserInput(): Handles validated user input.

spellingPractice(): Facilitates the spelling practice sessions.

addWord() & deleteWords(): Manages the addition and deletion of words.

loadWords() & saveWords(): Handles file operations for persistence.

Utility Functions:

clearConsole() & pauseConsole(): Manage console behavior.

CentredMessage() & rlines(): Enhance UI with centered text and separator lines.

[**See Full Code**](https://github.com/MIbnEKhalid/ICTLabProject/blob/main/index.cpp)

**Results**

The application successfully fulfills the following:

1. Allows users to practice spelling effectively.
2. Enables users to add, view, and delete words dynamically.
3. Provides a visually engaging CLI with color-coded feedback and centered messages.
4. Offers persistent data storage using file handling.

Users have reported a significant improvement in spelling skills after consistent use.

**References**

1.Project GitHub Repository: <https://github.com/MIbnEKhalid/ICTLabProject>

2. ANSI Escape Codes Documentation: <https://en.wikipedia.org/wiki/ANSI_escape_code>

3. C++ File Handling: https://cplusplus.com/doc/tutorial/files/