# Quiz Game Project PF – CCP Assignment

Date: 2025.11.10

By: Zainab Kinza Sheikh
Areefa Samar
Muzna Zubairy

Team: The Quizzards

# Contents

# Contents

```
========================================
        WELCOME TO MULTI-SUBJECT QUIZ GAME
========================================

Enter your name: Alina
Choose a Subject:
1. English
2. Mathematics
3. General Knowledge
4. Geography

Enter your choice (1-4): _
```

# Project Overview

## ✉ Objective of the Project

The project aims to develop an interactive quiz game that enhances learning through engagement, testing knowledge effectively while demonstrating programming skills and concepts in a practical application.

## ☑ Scope and Features of the Quiz Game

The Quiz Game project encompasses multiple-choice questions, lifelines (hints), score tracking, and user-friendly interfaces, designed to enhance learning engagement and assess knowledge effectively within the CCP curriculum.

## 🚩 Subjects Included: English, Maths, Geography, General Knowledge

The quiz game project covers core academic subjects—English, Maths, Geography, and General Knowledge—designed to enhance knowledge retention and critical thinking for students.

# Project Requirements and Tools

```c
void displayMenu() {
    clearScreen();
    printf(CYAN "\n-------------------------------------\n" RESET);
    printf(GREEN "           MCQ QUIZ PROGRAM\n" RESET);
    printf(CYAN "-------------------------------------\n" RESET);
    printf(YELLOW "1. Take Test\n" RESET);
    printf(YELLOW "2. View Leaderboard\n" RESET);
    printf(YELLOW "3. Exit\n" RESET);
    printf(CYAN "=====================================\n" RESET);
}

void takeTest() {
    int choice;
    char playAgain;
    Question questions[MAX_QUESTIONS];
    time_t start_time, end_time;
    LeaderboardEntry entry;
    const char *filename;

    do {
        clearScreen();
        printf(CYAN "\n-------------------------------------------------------\n" RESET);
        printf(GREEN "         WELCOME TO MULTI-SUBJECT QUIZ GAME         \n" RESET);
        printf(CYAN "-------------------------------------------------------\n\n" RESET);

        printf(LIGHT_CYAN "Enter your name: " RESET);
        fgets(entry.name, 50, stdin);
        entry.name[strcspn(entry.name, "\n")] = 0;

        printf(YELLOW "Choose a Subject:\n" RESET);
        printf(YELLOW "1. English\n2. Mathematics\n3. General Knowledge\n4. Geography\n" RESET);
        printf(LIGHT_CYAN "\nEnter your choice (1-4): " RESET);
        if(scanf("%d", &choice)!=1) {
            while(getchar()!='\n');
            printf(RED "Invalid input! Returning to menu.\n" RESET);
            return;
        }
        while(getchar()!='\n');

        switch (choice) {
            case 1:
```

## 01 Programming Language and Environment Used

The quiz game project utilizes .C due to its simplicity and extensive libraries, developed in Dev-C++ Embarcadero, providing a robust environment for efficient coding and debugging.

## 02 Tools and Software Requirements

The project requires the **C programming language** for development, using **Dev-C++** as the integrated development environment. Additionally, **Git** is used for version control, enabling efficient code management, collaboration, and seamless updates throughout the Quiz Game project.

## 03 Team Roles and Responsibilities

The team consists of a project manager, developer and tester, each ensuring timely delivery, functional gameplay, intuitive UI, and thorough quality assurance for project success.

# Contents

# Program Structure and Flow

## User Interface Design Overview

The user interface features intuitive navigation, clear visual hierarchy, and responsive design, enhancing user engagement and ensuring seamless interaction throughout the quiz game experience.

## Question & Scoring System

The quiz presents a structured set of questions for each subject and keeps an accurate record of the user's score, ensuring consistent evaluation and clear performance tracking throughout the game.

## Special Feature – Lifelines

The program offers three lifelines (hints) per subject to help users with difficult questions, making the quiz more engaging and interactive.

# Code Implementation Highlights

**Screenshot and Explanation of Main Program Code**

The main program code initializes game variables, handles user input, updates scores dynamically, and controls quiz flow, ensuring seamless interaction and efficient gameplay within the Quiz Game.

**Screenshot and Explanation of Subject-wise Question Code**

The subject-wise question code dynamically categorizes and retrieves questions based on topics, enhancing efficient quiz generation and ensuring targeted assessment aligned with subject learning objectives.

**Handling User Input and Validations**

User input is carefully validated to prevent errors and ensure accurate responses. Robust checks handle invalid entries, enhancing game reliability and improving overall user experience.

# Code Snippets

```
================================================
            MCQ QUIZ PROGRAM
================================================
1. Take Test
2. View Leaderboard
3. Exit
================================================
Enter your choice: 3
Thank you for using MCQ Program!
```

```
--------------------------------------------
Question 2 of 10
--------------------------------------------
Q2. Identify the figure of speech: "I must be cruel only to be kind." — Hamlet

A. Paradox
B. Irony
C. Metaphor
D. Oxymoron
Would you like a hint? (y/n): y
Hint: A statement that seems contradictory but expresses truth.


Enter your answer (A/B/C/D): a_
```

# Code Snippets

# Code Snippets

```
========================================
              QUIZ COMPLETED!
========================================
Your Total Score: 6 / 10
Hints Used: 3 / 3

Your score has been saved!

Do you want to play again? (y/n): _
```

```
========================================
              MCQ QUIZ PROGRAM
========================================
1. Take Test
2. View Leaderboard
3. Exit
========================================
Enter your choice: 3
Thank you for using MCQ Program!
```

# Contents

# Testing Methodology
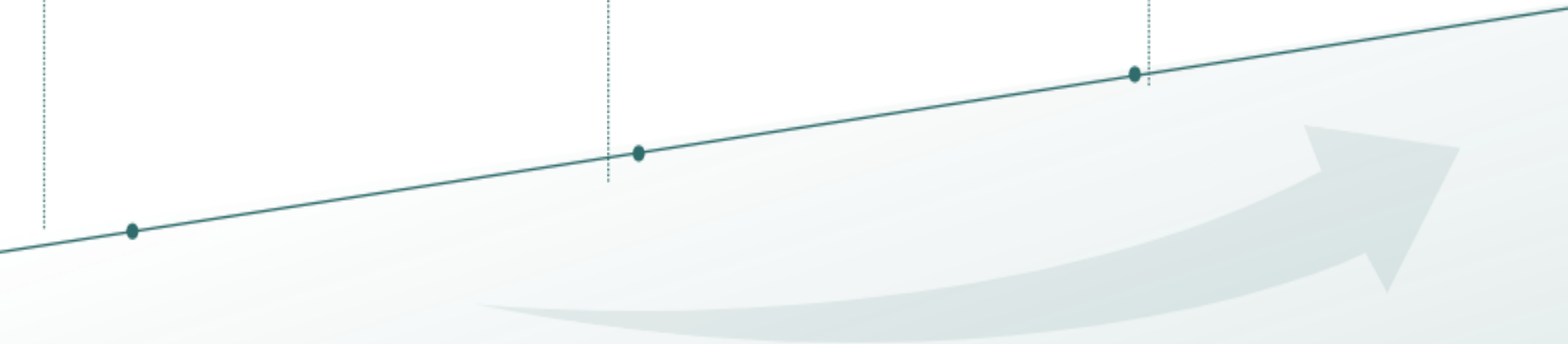
### Functional Testing of Quiz Features

Functional testing ensured all quiz features—question display, hints count, scoring, and answer validation—operate correctly, enhancing reliability and user experience by identifying and resolving key functional issues.

### Usability Testing

Usability testing revealed intuitive navigation and engaging gameplay, while feedback highlighted minor UI improvements, ensuring enhanced user experience and overall functionality alignment with project goals.

### Debugging and Improvements

Thorough debugging identified key logic flaws and UI issues, enabling targeted improvements that enhanced game stability, user experience, and overall performance efficiency throughout development.

# Challenges Faced

- **Technical Challenges During Development**

- Integrating diverse question formats and ensuring real-time score updates posed significant technical challenges, requiring robust coding and efficient debugging to maintain seamless gameplay and user experience.

- **Time Management and Coordination Issues**

- Time management difficulties and coordination lapses hindered progress, causing delays and misaligned task execution. Improved scheduling and communication strategies are essential to enhance team efficiency and project outcomes.

# Contents

# Project Summary and Learning Outcomes

**01**

### Achievements and Key Takeaways

Successfully developed an engaging quiz game, enhancing programming skills and teamwork. Key takeaways include problem-solving, effective collaboration, and practical application of theoretical concepts.

**02**

### Skills Gained Through the Project

Developed critical problem-solving, programming, and project management skills while enhancing teamwork and adaptability, laying a strong foundation for future complex program development projects.

# Future Improvements

## Step1

### Adding More Subjects and Questions

Improving the user interface with intuitive design and adding features like adaptive difficulty and multiplayer modes will enhance engagement and provide a more dynamic, user-centered quiz experience.

## Step2

### Multilingual Display

Introduce support for multiple languages, allowing users to take quizzes in their preferred language. This will make the game more inclusive and accessible to a wider audience.

## Step3

### Timed Mode

Introduce a visible timer and set a time limit for each question to boost engagement and add a sense of challenge to the quiz.

## Step4

### Immediate Sound Feedback

Integrate sound effects for correct and incorrect answers to provide instant feedback, making the quiz more dynamic and immersive.

## Step5

### Enhanced User Interface

Upgrade the interface with improved color themes, clear typography, and intuitive menus. Adding animations and better layout organization will make the quiz visually appealing and easier to navigate.

Thanks