Software Performance Report - Advanced Tic Tac Toe Game

1. Introduction

This document provides a detailed report on the performance monitoring implementation in the Advanced Tic Tac Toe Game. It focuses on runtime efficiency, memory consumption, and CPU utilization using custom instrumentation within the PerformanceMonitor class.

2. PerformanceMonitor Overview

The PerformanceMonitor class is responsible for tracking execution times of various game operations and collecting system-level statistics like memory and CPU usage. It is implemented in both the header and source files.

Key Capabilities:

- - Measure execution time (ms) using **startMeasurement()** and **stopMeasurement()**
- - Calculate average, max, and min execution times via **getAverageTime(), getMaxTime(),** and **getMinTime()**
- - Save measurement data using saveToFile()
- - Report memory usage using **getCurrentMemoryUsageMB()**
- - Report CPU usage using **getCPUUsagePercent()** (Windows only)

3. Measurement Methods

3.1 Execution Timing

Used to profile time-sensitive functions and operations including:

- - MainWindow::signInButtonClicked()
- - MainWindow::signUpButtonClicked()
- - DatabaseManager::saveUser() and DatabaseManager::verifyUser()
- - GameBoard::aiMove()

3.2 Summary Metrics

Performance summaries are retrieved from:

- - getAverageTime()
- - getMaxTime()

- - getMinTime()

3.3 File Logging

Measurement summaries are written to external reports using:

- - saveToFile()

4. Memory Usage Monitoring

Function: getCurrentMemoryUsageMB()

Class: PerformanceMonitor

Purpose: Monitors RAM consumption by the application during runtime, especially during resource-heavy activities like game history replay or AI computation.

5. CPU Usage Monitoring

Function: **getCPUUsagePercent()**

Class: PerformanceMonitor

Purpose: Estimates the percentage of CPU time utilized by the system while the game is running. This is crucial to evaluate AI load and database operation impact.

Note: This function is implemented using Windows API and is only applicable on Windows platforms.

6. Static Variables (CPU Tracking)

To calculate changes in system activity over time, these static variables are used:

- - lastIdleTime
- - lastKernelTime
- - lastUserTime

These track previous CPU state between measurements to calculate the usage percentage accurately.

7. Practical Usage in Project

- These timing functions are used for: Database operations (saving/loading users andgamehistory)Login operationsAI decision making Game duration

8. Conclusion

The integration of the PerformanceMonitor class into the Advanced Tic Tac Toe Game provides detailed insight into the app's efficiency and helps ensure smooth and responsive gameplay. By monitoring execution time, memory usage, and CPU load, the development team can continually refine system performance.