

Performance Measurement and Optimization

1. response time

- This part aims to analyze and document the response times of key interactions within the tic-tac-toe application. Efficient response times are crucial for user satisfaction, especially in interactive applications like games. This analysis will help identify any performance bottlenecks and provide recommendations for optimization.

2. Data Collection

- Timestamps were recorded at the start and end of each action, and the difference was calculated to determine the response time. The following actions were tracked :
- **Button Clicks:** Time taken from when a user clicks a button to when the application responds.
- **AI Move Calculation:** Time taken for the AI to calculate and make a move using the three levels functions.

Metrics

- 1) **Average Response Time:** The mean time taken for each action.
- 2) **Maximum Response Time:** The longest time recorded for an action.
- 3) **Minimum Response Time:** The shortest time recorded for an action.

3. Data Analysis

- Data was collected over multiple gameplay sessions, with a total of 100 button clicks in the two players mode and 50 in each AI level moves recorded.

Action	Average time response	Max time response	Min time response
Button click in two players mode	2msec	27msec	0
AI move(easy level)	5msec	25msec	0
AI move(medium level)	6msec	27msec	0
AI move(hard level)	104msec	400msec	0

- The response times for button clicks in the two players mode and in the easy and medium AI modes are within an acceptable range for a smooth user experience.
- The hard AI move calculation times are longer and show more variability. The maximum response time of 400ms indicates potential performance bottlenecks during complex game states. This can affect the smoothness of gameplay, especially in more complex game states.

4. Recommendations to improve time response

1. **Implement Multi-threading:** Use multi-threading to perform AI calculations in the background, preventing the UI from becoming unresponsive.
2. **Cache Results:** Implement caching for repeated game states to reduce calculation times.