

QuizMaster

A poll-style quiz game for
classroom use

By Philip Smith

Overview

- QuizMaster is a client-server based poll program similar to the “Poll the audience” lifeline from the popular game show “Who Wants to be a Millionaire.”

Design Considerations

- Language Choice
 - Visual Basic
 - HTML forms
 - Java
 - C++
 - Scripted front-end with Java core components

Design Considerations

- Networking considerations
 - Number of players
 - Data to send from client to server
 - Collisions and dropped inputs
- Other considerations
 - Must be graphical
 - Quiz file format

Implementation

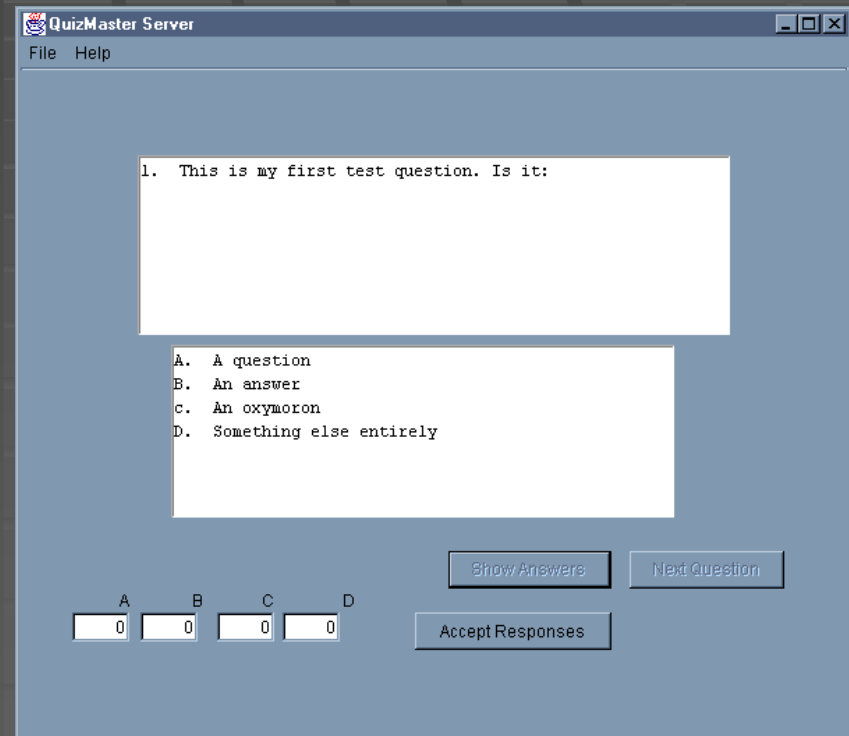
- Tools
 - Began with IBM VisualAge
 - Moved to Borland's JBuilder 5
- Separate client and server components
- Simple GUI creation
- Quiz file implementation
 - Newline (\n) delineated
 - Entire question or answer must fit on single line

Sample quiz

- 1. This is a sample question
- A. This is answer A
- B. This is answer B
- C. This is answer C
- D. This is answer D
- 2. This is a second sample question
 - ...

QuizMaster Server

- The server administrates the quiz by loading the quiz file, displaying answers, and displaying collected responses for each question.



The screenshot shows the QuizMaster Server application window. The title bar reads "QuizMaster Server" with standard window controls. The menu bar contains "File" and "Help". The main content area displays a question: "1. This is my first test question. Is it:". Below the question is a list of four options: "A. A question", "B. An answer", "C. An oxymoron", and "D. Something else entirely". At the bottom of the window, there are four input fields labeled A, B, C, and D, each containing the number "0". To the right of these fields are three buttons: "Show Answers", "Next Question", and "Accept Responses".

QuizMaster Server

File Help

1. This is my first test question. Is it:

A. A question
B. An answer
C. An oxymoron
D. Something else entirely

A B C D

0 0 0 0

Show Answers Next Question

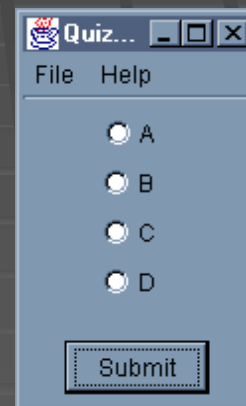
Accept Responses

Server difficulties

- Problem: Reading the questions/answers from the input file
- Solution: BufferedReader class allows for line-by-line reading of a text file.
- Problem: Update speed not allowing answers to display before waiting for responses
- Solution: Added third button to GUI to begin accepting answers

QuizMaster Client

- Originally displayed questions also.
 - Too much information to transfer
- Client is straightforward
 - Radio buttons to select response
 - Button to submit answer to server



Networking Model

- Sockets
- Streams
- True client server vs. connectionless

Networking Model

- Multithreaded

- Separate thread and socket for each client
- Forced to wait for input from clients in sequential order
- Conveyor belt analogy

- Datagrams

- Single socket accepts inputs from all clients
- Accepts multiple inputs simultaneously
- Small packets
- Mailbox analogy

The Future of QuizMaster

- Networking class, Spring 2002
- Continuing updates to software
 - Server properties dialog allowing user to select number of players, as well as port settings
 - Client connection properties dialog allowing user to specify IP address to connect to, port settings, etc.
 - “On the fly” question and answer generation

What have we learned?

- True application design is a BIG task
- Need clear plan of attack
- Not just coding
- Proper tools

Questions?