

JOGO DA VELHA

EDIÇÃO MULTIPLAYER



UNIVERSIDADE FEDERAL DE SERGIPE
ALUNOS: VERENILSON DA SILVA SOUZA
GUILHERME ANDRADE DE MEDEIROS

Índice

1. Introdução ao Jogo da Velha;

2. Características gerais do produto;
3. Implementação do software;
 - 3.1. Client-side;
 - 3.2. Server-side;
 - 3.3. Interface Gráfica;
4. Comunicação do multiplayer;
5. Conclusões.

Capítulo 1: Introdução ao Jogo da Velha

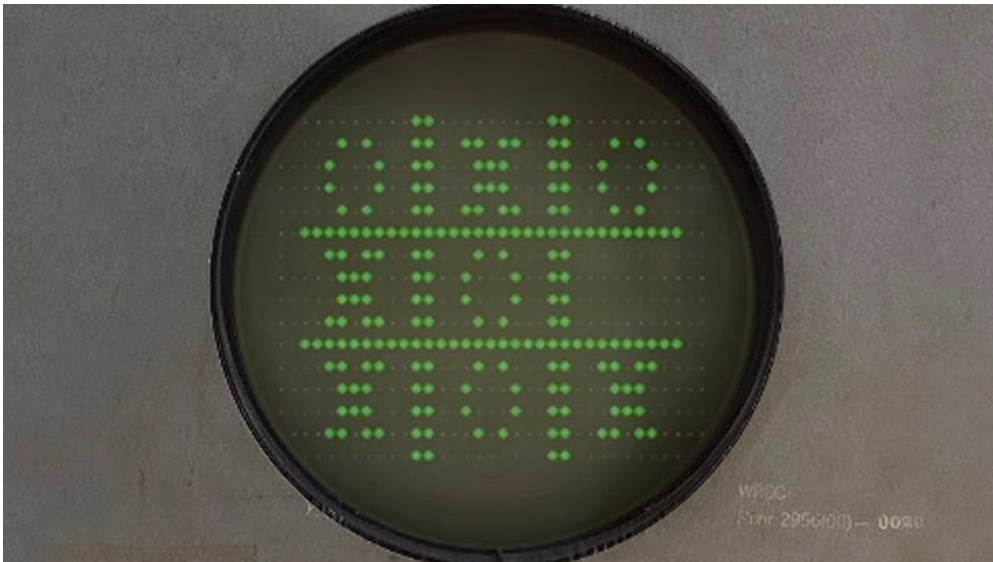
A origem do jogo da velha é incerta, mas acredita-se que ele exista há milhares de anos. Evidências arqueológicas sugerem que o jogo era conhecido no antigo Egito, Roma e China. No entanto, não há consenso sobre qual cultura o inventou primeiro. Algumas das primeiras representações do jogo da velha foram encontradas em tabuleiros esculpidos em pedras no Egito, datados de 3.500 anos atrás. Tabuleiros semelhantes também foram encontrados em escavações romanas e chinesas.

Ao longo dos séculos, o jogo da velha se espalhou pelo mundo e se tornou um passatempo popular em muitas culturas. O nome do jogo varia de acordo com o país. No Brasil, é conhecido como jogo da velha, enquanto em Portugal é chamado de jogo do galo.

O jogo da velha é um jogo simples, mas desafiador, que pode ser jogado por pessoas de todas as idades. É um ótimo jogo para desenvolver habilidades de raciocínio lógico e estratégia.

Nao é recente...

- Em 1952, foi desenvolvido o jogo OXO para computador EDSAC, onde o jogador desafiava o computador em partidas de Jogo da



Velha. Assim, surgia um dos primeiros jogos de vídeo game do qual se tem notícia.

Capítulo 2: Sobre a Implementação

Nosso jogo foi meticulosamente criado para proporcionar uma **experiência multijogador envolvente e dinâmica**. Através de uma arquitetura robusta e escalável, você poderá se conectar com amigos e jogadores do mundo todo, participando de partidas emocionantes e inesquecíveis.

Tecnologias de ponta impulsionam a sua experiência:

- **Desenvolvido em .NET e C#:** Tecnologias *mainstreams* e modernas que garantem alto desempenho e confiabilidade.
- **Escalabilidade:** O jogo se adapta à demanda, permitindo que mais jogadores participem sem comprometer a qualidade da experiência.

Capítulo 3: Implementação do Software

Nosso jogo foi desenhado com uma estrutura pensada na escalabilidade e praticidade, há basicamente duas estruturas principais, o servidor e o cliente, ao trabalhar de forma independente é possível alcançar um melhor desempenho e tornar os comandos mais efetivos e precisos.

- **Cliente:** Envia seus comandos e te coloca na ação, exibindo tudo o que acontece na tela com agilidade e precisão.
- **Servidor:** Cria e gerencia as partidas, garantindo que tudo funcione perfeitamente para todos os jogadores.

Agora, vamos falar das classes que representam a estrutura do **cliente**:

- Classe Program.cs:
 - Cria um aplicativo cliente para um jogo da velha que se conecta a um servidor na mesma máquina especificada, podendo ser local ou remoto. Ele tenta estabelecer uma conexão de rede e inicia uma thread separada para lidar com a comunicação e o gerenciamento do jogo.
- Classe GerenciarPartida.cs:
 - Este código implementa a lógica de gerenciamento de partidas para o jogo que se conecta ao servidor. Ele lida com a comunicação com o servidor, estabelece a partida com outro jogador e inicia a thread para gerenciar o jogo.
- Classe IdJogador.cs:
 - Esta classe funciona como uma ponte de comunicação entre o aplicativo cliente e o servidor de jogo, permitindo que o jogador envie e receba mensagens, gerencie seu status no jogo e participe do Jogo da Velha.

- Classe MatchCliente.cs:
 - Ela é responsável por gerenciar o jogo do lado do cliente no Jogo da Velha. Ela se comunica com o servidor para receber atualizações da partida, imprime o estado atual do jogo na tela, permite que o jogador faça jogadas e anuncia o vencedor ou derrotado quando a partida termina.

Em resumo, as classes Jogador e MatchCliente trabalham em conjunto para gerenciar a comunicação entre o cliente e o servidor, permitir que os jogadores façam jogadas e acompanhar o estado do jogo. A classe Jogador lida com as informações do jogador e a comunicação básica com o servidor, enquanto a classe MatchCliente gerencia o fluxo da partida, a atualização da matriz do jogo, a exibição de informações na tela e a interação com o jogador.

Por FIM, vamos falar sobre as classes que compoem a estrutura do Servidor, parte fundamental para a interação entre os jogadores.

- Classe Program.cs:
 - Esta classe define a estrutura básica para um servidor do Jogo da Velha. Ela inicia o servidor, aceita conexões de clientes e passa as conexões para gerenciar as partidas.
- Classe GerenciadorPartidas.cs:
 - Esta classe fornece funcionalidades básicas para gerenciar partidas de Jogo da Velha em no servidor, ela utiliza threads para tratar as conexões dos jogadores de forma assíncrona, além de ter um sistema de emparelhamento de jogadores.
- Classe IdJogador.cs:
 - Esta classe possui as informações e funcionalidades necessárias para se comunicar com um jogador conectado ao servidor. Ela permite enviar e receber mensagens de

texto, identificando o jogador pelo seu nome e indicando qual jogador possui a vez no início da partida.

- Classe Partida.cs:
 - Esta classe é responsável por gerenciar uma partida entre dois jogadores conectados ao servidor. Algumas de suas responsabilidades inclui iniciar a partida com os nomes informados pelos jogadores, controlar a vez de cada jogador, validar as jogadas, atualizar a matriz do jogo que representa o tabuleiro, verificar se a partida acabou ou empatou e quem é o ganhador, finalizar a partida e encerrar a conexão.

Capítulo 4: Multiplayer

O sistema de comunicação multiplayer é implementado de forma eficiente e robusta, utilizando sockets TCP, threads, protocolos de comunicação personalizados e mecanismos de sincronização. Na versão com interface gráfica, é requisitado o IP e Porta que será usado para conectar ao servidor, isso permite muito mais flexibilidade que retira a necessidade de alterar o código fonte do jogo toda vez que o IP do servidor mudar.

Capítulo 5: Conclusões

O projeto Jogo da Velha demonstra uma implementação funcional de um jogo clássico em um ambiente multiplayer online. A arquitetura do servidor utiliza conceitos adequados de programação orientada a objetos, incluindo classes, métodos, threads e encapsulamento. A comunicação entre o servidor e os jogadores é bem estruturada e eficiente, utilizando sockets TCP, threads dedicadas e um protocolo de comunicação personalizado. Há muito espaço para melhorias, como maior escalabilidade, um gerenciamento de erros mais adequado, segurança e

mais funcionalidades, mas acreditamos que o projeto está em um nível de maturidade bastante satisfatório e equilibrado com as demandas que ele pretende cumprir.