

# Projeto Magic Trick: Jogando de Forma Autônoma

Phelipe Pereira de Souza

Bacharelado em Ciência da Computação

Centro Universitário SENAC - Campus Santo Amaro (SENAC-SP)

Av. Engenheiro Eusébio Stevaux, 823 – Santo Amaro, São Paulo – CEP 04696-000 – SP – Brasil

## Introdução

Magic Trick é um jogo de cartas, não muito conhecido, para se jogar com os amigos. O objetivo desse projeto é criar um algoritmo que possa jogar de forma autônoma, ou seja por conta própria.

Figura: Naipes das cartas *Magic Trick*



Valores das cartas de 0 a 7 cada naipe

Basicamente, as regras do jogo são:

- ▶ 14 cartas para cada jogador (recomendado 4 jogadores).
- ▶ Alguém que não estiver jogando deve embaralhar e distribuir as cartas de forma que ninguém possa ver qual o valor, apenas o seu naipe. As cartas devem ficar na frente de cada jogador ordenadas de forma crescente.
- ▶ Todas as cartas jogadas devem ter seu valor revelado. Quem começar deve jogar uma carta de qualquer naipe. Já os outros jogadores devem jogar uma carta com o mesmo naipe.
- ▶ Somente quem não tiver o mesmo naipe pode jogar qualquer outra carta, todos os naipes diferentes perdem, menos o de copas, ele ganha dos outros naipes.
- ▶ Vence a rodada aquele que jogou uma carta de copas com o maior valor, caso nenhuma carta de copas tenha sido jogada ganha aquele que respeitou o naipe de abertura e com o maior valor. O vencedor da rodada atual deve começar a próxima rodada.
- ▶ Durante uma das rodadas os jogadores devem realizar uma aposta obrigatória. A aposta significa quantas rodadas o jogador acha que vai vencer, o número da aposta é dado a partir de uma carta, a escolha do jogador, das cartas que ele ainda não jogou.
- ▶ Quando todas as cartas forem jogadas, é a hora de verificar as apostas. Aqueles que apostaram 1 ou 5 porém venceram 3 rodadas perdem -2 pontos na sua pontuação final. Aqueles que acertaram suas apostas, ou seja apostaram 5 e ganharam 5 rodadas ganham mais 3 pontos na sua pontuação final.
- ▶ Depois de verificar as apostas, todas as cartas devem ser embaralhadas e distribuídas da mesma maneira do início do jogo. Agora quem deve começar é o jogador a esquerda de quem começou a partida passada.
- ▶ O jogo só acaba quando todos os jogadores tiverem começado 1 vez a partida. Ou seja, se tiver 4 jogadores, o jogo deverá ter 4 partidas.
- ▶ Quando o jogo acabar, o vencedor é aquele que teve a maior pontuação final.

## Metodologia

Antes de começar o desenvolvimento do algoritmo, foi necessário entender as regras do jogo e como elas funcionavam na prática, para isso foi utilizado um aplicativo chamado *Trickster's Table*, para celular, que já continha esse jogo.

Após baixar o aplicativo, foram jogadas aproximadamente 30 partidas, sendo 10 para entender as regras e como tudo funcionava, e outras 20 para encontrar algum padrão nas jogadas que ajudasse a formar uma lógica.

Depois de algumas partidas foi possível anotar os padrões que se repetiam e formar uma estratégia "base" no papel.

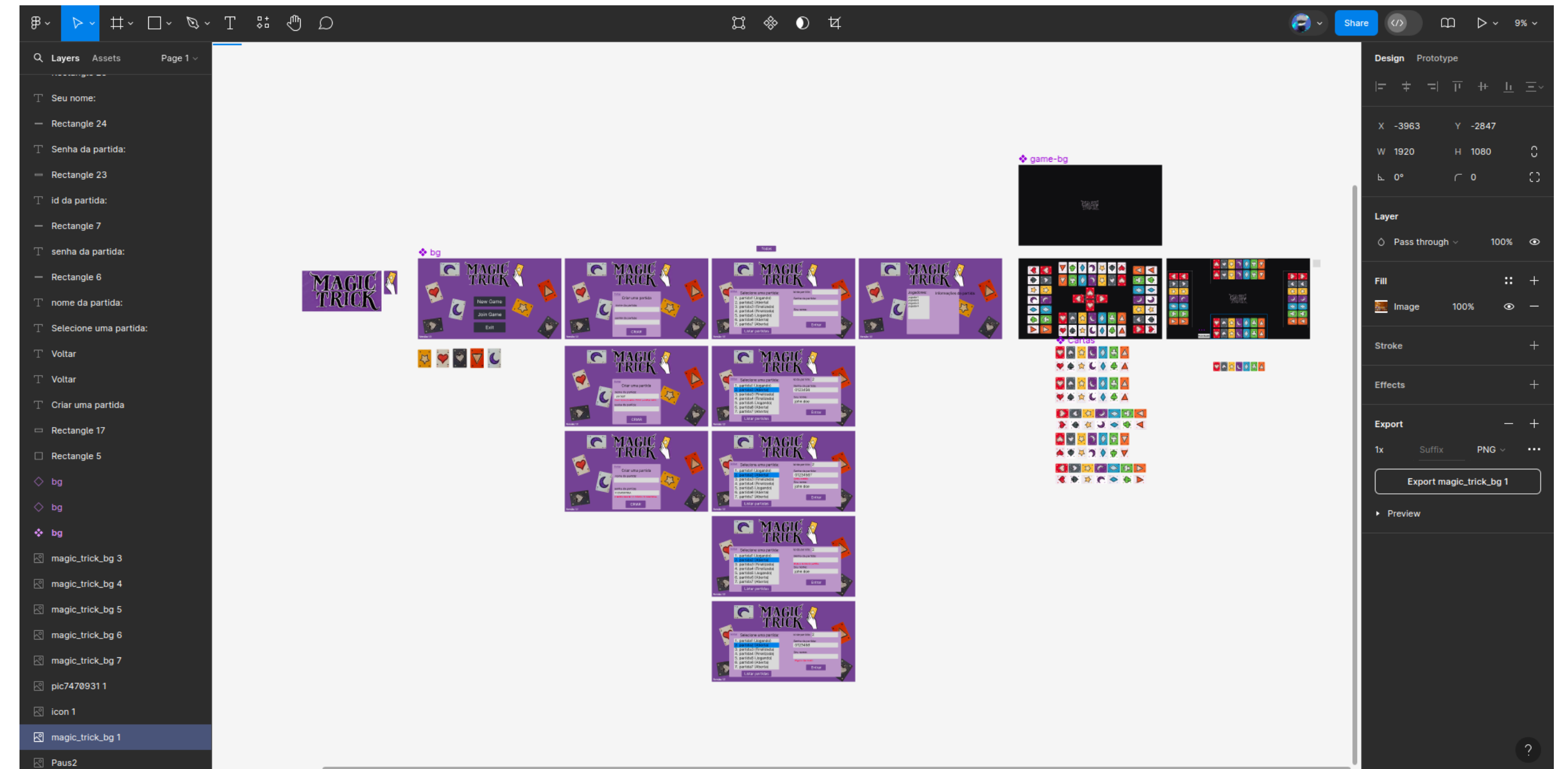
## Tecnologia

O trabalho foi feito utilizando a linguagem de programação C# e a IDE Visual Studio.

## Desenvolvimento

Antes de começar o desenvolvimento da interface do projeto, foi utilizado o *Figma* para ter uma visão mais clara de como tudo deveria ficar e funcionar.

Figura: Protótipo de baixa fidelidade



Como tudo já estava definido no Figma, foi muito mais fácil na hora de realmente fazer a interface do jogo, pois o Visual Studio permite que você posicione os componentes do seu programa por Design Time, ou seja, não foi necessário definir a posição dos botões, janelas e onde as cartas devem ficar pelo código.

A parte mais complicada, além de montar uma boa estratégia, foi aplicar os conceitos de Programação Orientada a Objetos (POO), pois inicialmente não havia um conhecimento sólido sobre esse paradigma. Após algumas lições durante o desenvolvimento, o paradigma de POO foi ficando mais claro e por conta disso houve diversas refatorações no código pelo caminho.

## Resultado

Depois de aplicar, dentro do código, a estratégia encontrada na metodologia e finalizar toda a interface planejada pelo Figma. Pode-se verificar o resultado a seguir.

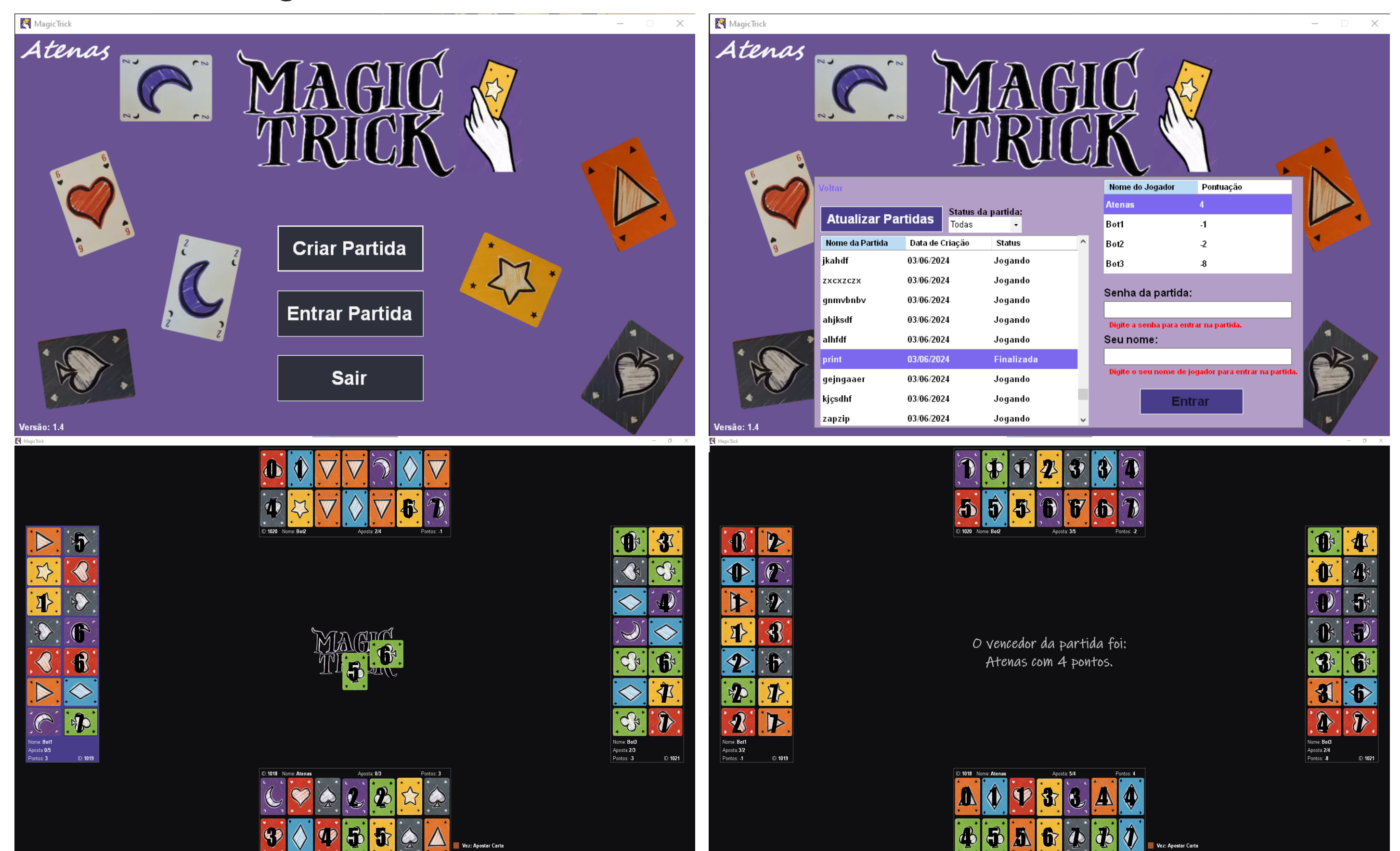


Figura: Estratégia planejada

### Estratégia

**Aposta:** Sempre apostar a oitava carta ou a carta mais ao meio possível.

#### Preciso ganhar?

- Se eu começo, então joga maior carta.
- Se não tiver o naipe de abertura, então joga a carta de copas mais baixa.
- Se não tiver o naipe de abertura e não tiver carta de copas, então joga carta mais baixa.
- Se tiver o naipe de abertura, então joga o que parecer ter o maior valor.

#### Preciso perder?

- Se eu começo, então joga menor carta.
- Se não tiver o naipe de abertura, então joga a carta mais alta, desde que ela não seja de copas.
- Se não tiver o naipe de abertura e só tiver carta de copas, então joga a carta mais baixa.
- Se tiver o naipe de abertura, então joga o que parecer ter o menor valor.

O algoritmo encontrado como solução foi a estratégia na figura acima.