

# Especificação de Requisitos do Cho

## Projeto do jogo Cho

### Especificação de requisitos de software

---

Versão	Autores	Data	Ação
1.0	Samuel Vieira Bernardo Mateus Arns Kreuch Vitor Lorizete de Lima Filho	22/11/2021	Estabelecimento dos requisitos de software
2.0	Samuel Vieira Bernardo Mateus Arns Kreuch Vitor Lorizete de Lima Filho	31/01/2022	Refinamento dos requisitos de software
3.0	Samuel Vieira Bernardo Mateus Arns Kreuch Vitor Lorizete de Lima Filho	02/03/2022	Modelagem
4.0	Samuel Vieira Bernardo Mateus Arns Kreuch Vitor Lorizete de Lima Filho	14/03/2022	Trabalho completo

#### Conteúdo:

1. Introdução
2. Visão geral do Sistema
3. Requisitos da Aplicação
4. Esboço da interface gráfica

## 1. Introdução

---

**Objetivo:** desenvolvimento de um jogo multiplayer, em que dois usuários jogam em turnos (de maneira competitiva), onde o objetivo é matar a rainha inimiga ou chegar no centro com a sua própria rainha e permanecer lá durante um turno.

**O jogo:** Jogo de tabuleiro, o qual é circular, composto de vários anéis. Em algumas áreas do tabuleiro, não há caminho direto indo de um anel exterior para um mais

interior; essas áreas são chamadas de intersecções. Existem três tipos de peças; peões, que andam de um em um, e são a única peça capaz de atravessar intersecções (porém só em direção ao centro); torres, que andam infinitos pisos; e rainhas, que andam de dois em dois, e não conseguem voltar (isto é, ir de um anel interior para um mais exterior). Peões e torres ao chegar ao centro do tabuleiro mudam de profissão de um para outro; peões se transformam em torres e torres se transformam em peão. Por último, os anéis do tabuleiro podem girar, reposicionando as peças em relação aos outros anéis. Isto ocorre em duas ocasiões: dois movimentos são feitos na mesma direção (horário ou anti-horário) no mesmo anel; e ao uma rainha se movimentar duas casas para os lados.

**Descrição completa do jogo:** Cho é um jogo de tabuleiro de estratégia baseado em turnos, onde duas pessoas jogam simultaneamente uma contra a outra, onde cada jogador tem um número inicial fixo de 10 peças, e a partir da movimentação dessas peças no tabuleiro (o tabuleiro será melhor explicado logo em seguida, porém podemos ver um exemplo dele na Figura 2) é possível derrotar as peças do adversário e vencer a partida, o objetivo final do jogo é derrotar a peça adversária denotada como rainha(os tipos de peças serão explicados logo em seguida) ou alcançar o centro do tabuleiro com a sua rainha e permanecer lá por um turno (Figura 5).

Sobre o tabuleiro: O tabuleiro é em forma circular e dividido em 3 anéis de tamanhos diferentes; esses anéis tem grande importância em nosso jogo, pois além de ser a partir deles que as peças se movimentam, os mesmos tem a mecânica de poderem girar e com isso reposicionar as peças, gerando assim mais estratégias para o jogo. Para que os anéis girem, é preciso que dois movimentos sejam feitos na mesma direção (horário ou anti-horário) no mesmo anel; ou ao uma rainha se movimentar duas casas para o lado.

Sobre as peças: As peças são classificadas entre três tipos, sendo elas peões, torres e rainhas. Cada jogador inicia a partida com um total de 10 peças, 7 peões, 2 torres e 1 rainha (Figura 6), cada peça tem seu estilo de movimentação, o peão pode mover uma casa apenas, sendo verticalmente ou horizontalmente (Figura 7) e ainda é o único que pode passar por uma intersecção (Figura 9) e caso ele chegue ao centro do tabuleiro ele se tornará uma torre, já a torre pode mover-se infinitamente verticalmente ou horizontalmente (Figura 7.1) e caso ela chegue ao

centro do tabuleiro, ela se tornará um peão, e por fim a rainha pode mover-se até 2 casas verticalmente ou horizontalmente (Figura 7.2), porém nunca retrocedendo.

## 2. Visão Geral do Sistema

---

### 2.1 Arquitetura do software

- Programa orientado a objetos (centralizado).

### 2.2 Premissas de desenvolvimento

- Programa deve possuir interface gráfica bidimensional;
- Programa deve ser implementado em linguagem Python;
- O jogo ocorre com 2 jogadores (jogando de maneira competitiva).
- O projeto deve ser produzido com Visual Paradigm.

## 3. Requisitos da Aplicação

---

### 3.1 Requisitos Funcionais

- **Requisito funcional 1 - Iniciar partida:** o programa deve fazer a renderização dos arcos para a geração do tabuleiro (figura 2), juntamente com a renderização das peças nas devidas posições (figura 3);
- **Requisito funcional 2 - Validação da jogada/Mover peças(Executar jogada):** O programa deverá seguir as regras descritas em “**Descrição completa do jogo, Sobre as peças:**”, juntamente com as validações (figura 7, 7.1, 7.2) impedindo que a jogada seja executada além da marcação descritas nas mesmas, as peças poderão ser movimentadas em colisão com as do inimigo, fazendo com que a mesma seja derrotada;
- **Requisito funcional 3 - Perder a partida:** Ao seguir as regras de movimentação descrita no **Requisito funcional 2**, caso sua rainha seja derrotada ou o adversário permanecer com sua rainha no centro do tabuleiro por um turno completo, você perderá a partida;
- **Requisito funcional 4 - Ganhar a partida:** Ao seguir as regras de movimentação descrita no **Requisito funcional 2**, caso sua rainha seja posta

no centro do tabuleiro durante um turno completo, ou uma de suas peças derrote a rainha adversária, você ganhará e deverá aparecer a tela de vitória (Figura 8, 8.1);

- **Requisito funcional 5 - Sair da partida:** a qualquer momento da partida, é possível executar o “EXIT” do jogo a partir da tecla “ESC”, acessando a tela de saída (Figura 4);

### 3.2 Requisitos Não Funcionais

- **Requisito não funcional 1 – Especificações do projeto:** além do código Python, deve ser produzida especificação de projeto baseada em UML 2;
- **Requisito não funcional 2 – Interface gráfica:** o programa deverá ter uma interface gráfica única, compartilhada pelos usuários que a utilizam;
- **Requisito não funcional 3 – Tecnologia de interface gráfica:** a interface gráfica deve ser também com a biblioteca Pygame;
- **Requisito não funcional 4 – Resolução do jogo:** o jogo deve ter uma resolução variável, desenhada de acordo com a tela necessária;
- **Requisito não funcional 5 - Linguagem de programação:** o jogo deve ser implementado em python.

## 4. Esboço da Interface Gráfica



Figura 1: Tela inicial do jogo (menu principal).

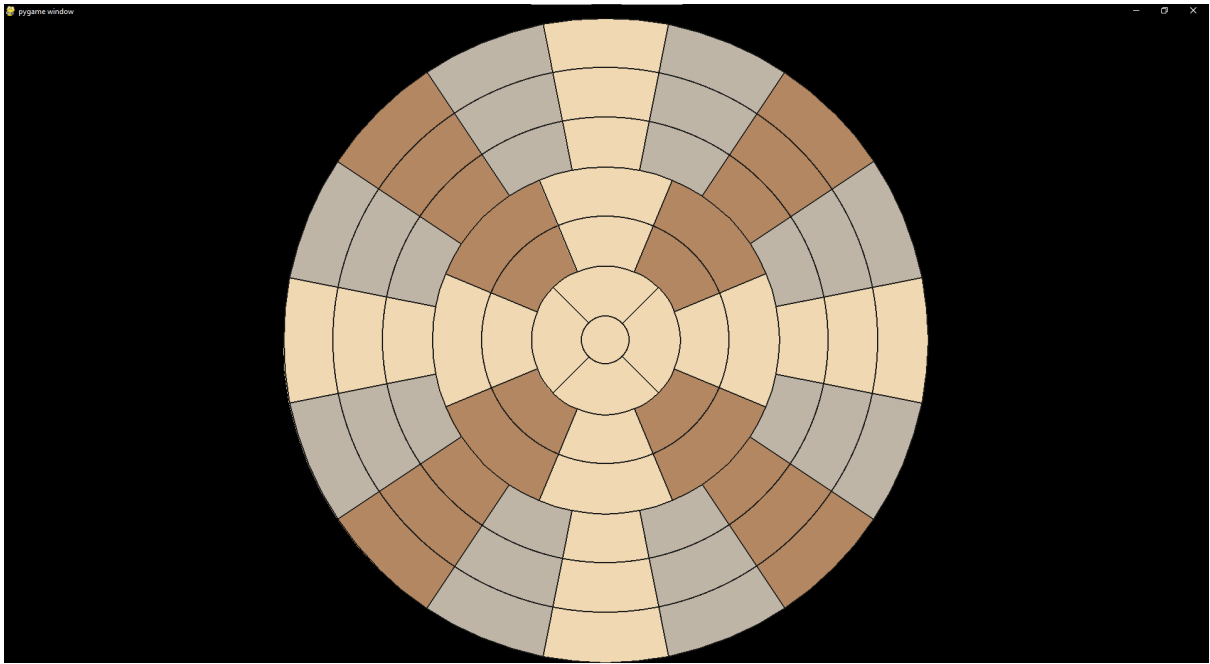


Figura 2: Figura do tabuleiro inicial, sem as peças.

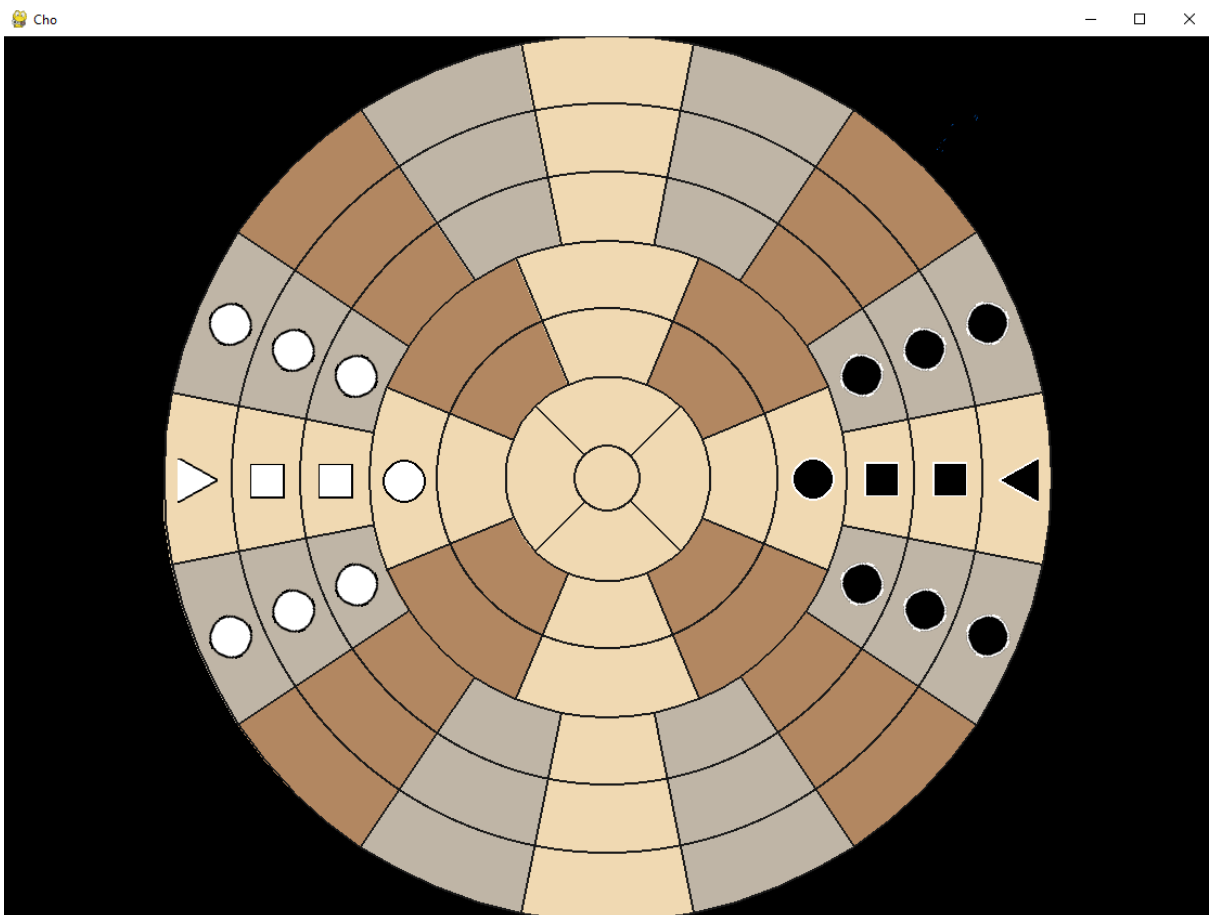


Figura 3: Figura do tabuleiro, com as peças..

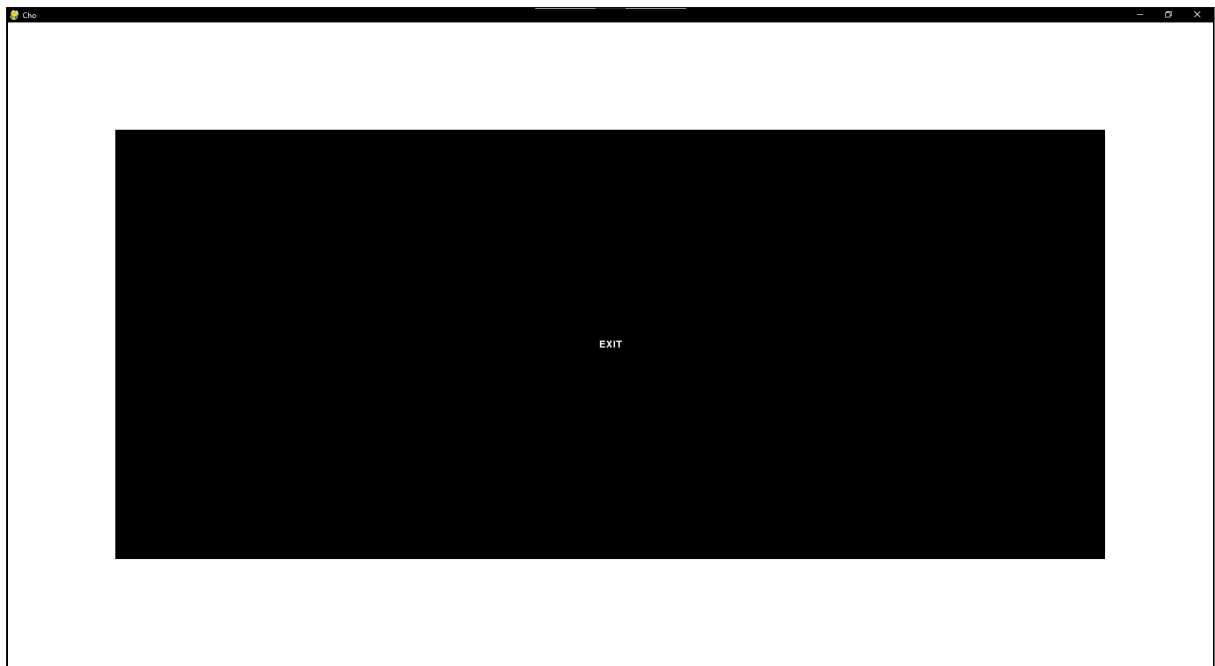


Figura 4: Menu de saída do jogo

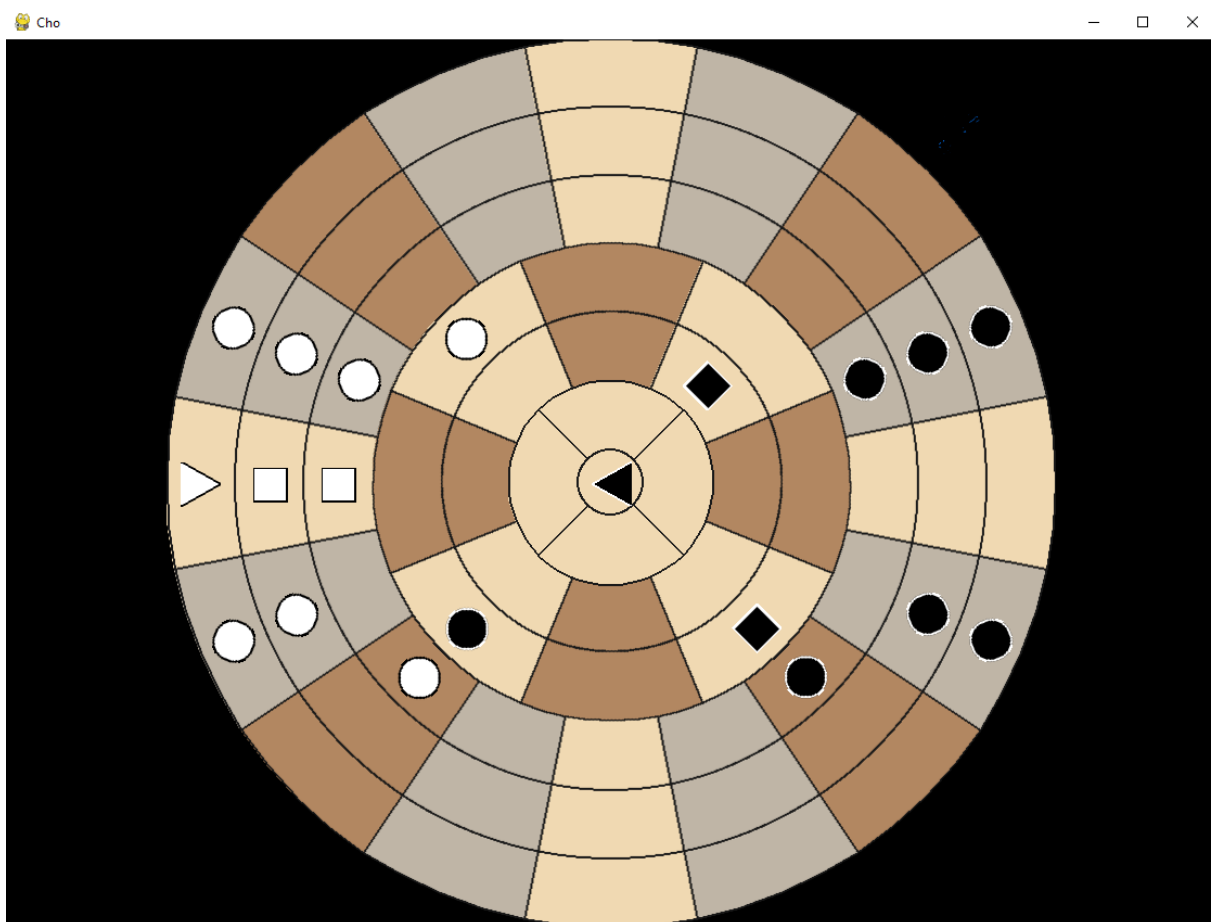


Figura 5: rainha no centro do tabuleiro

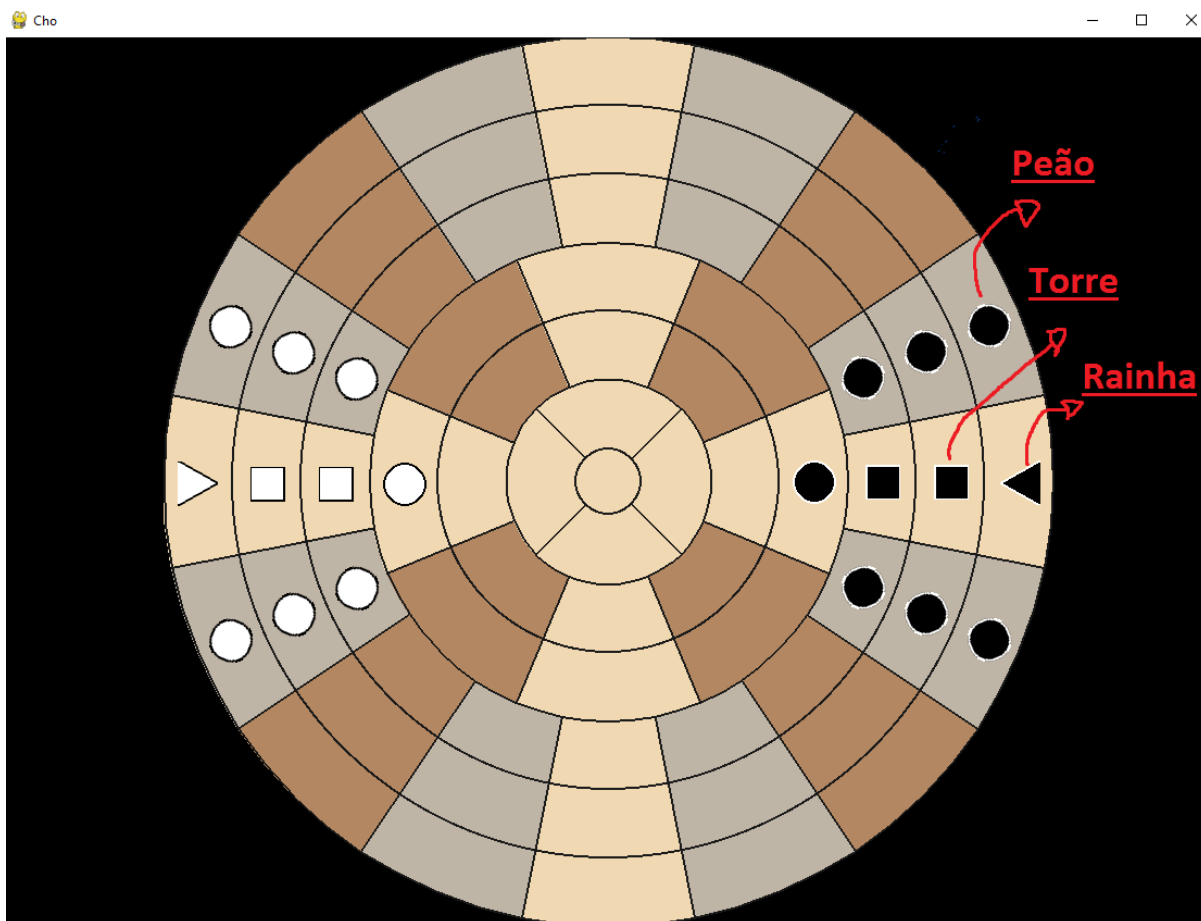


Figura 6: Referenciando as peças do jogo

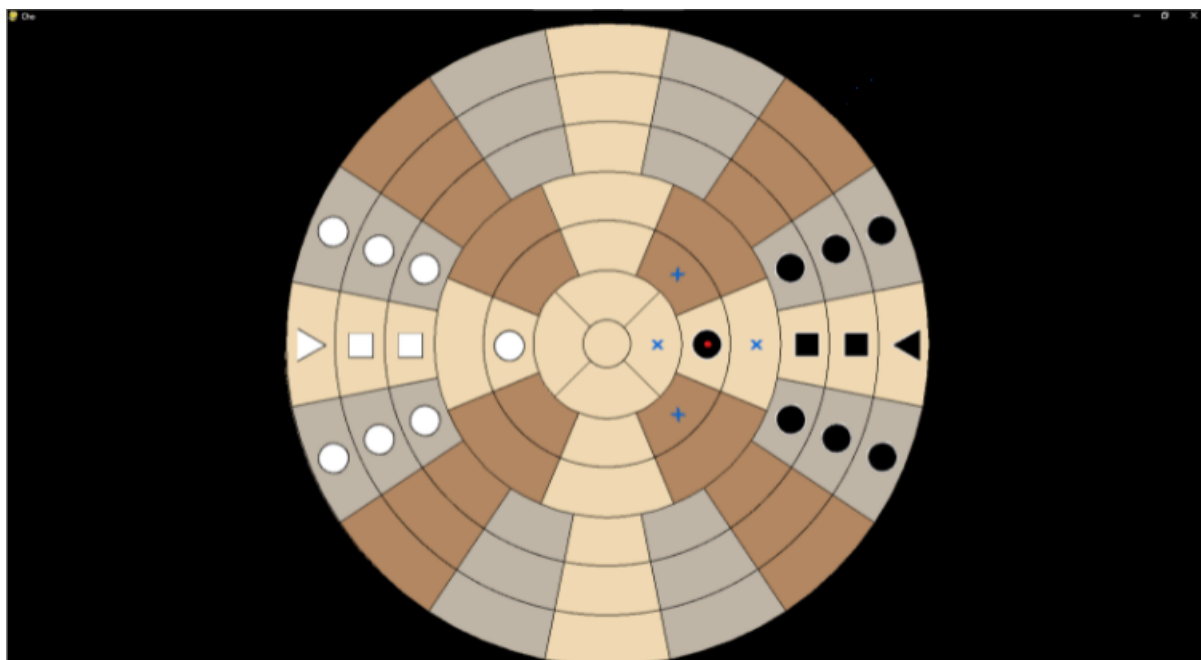


Figura 7: Referenciando as casas possíveis para a movimentação do Peão destacado

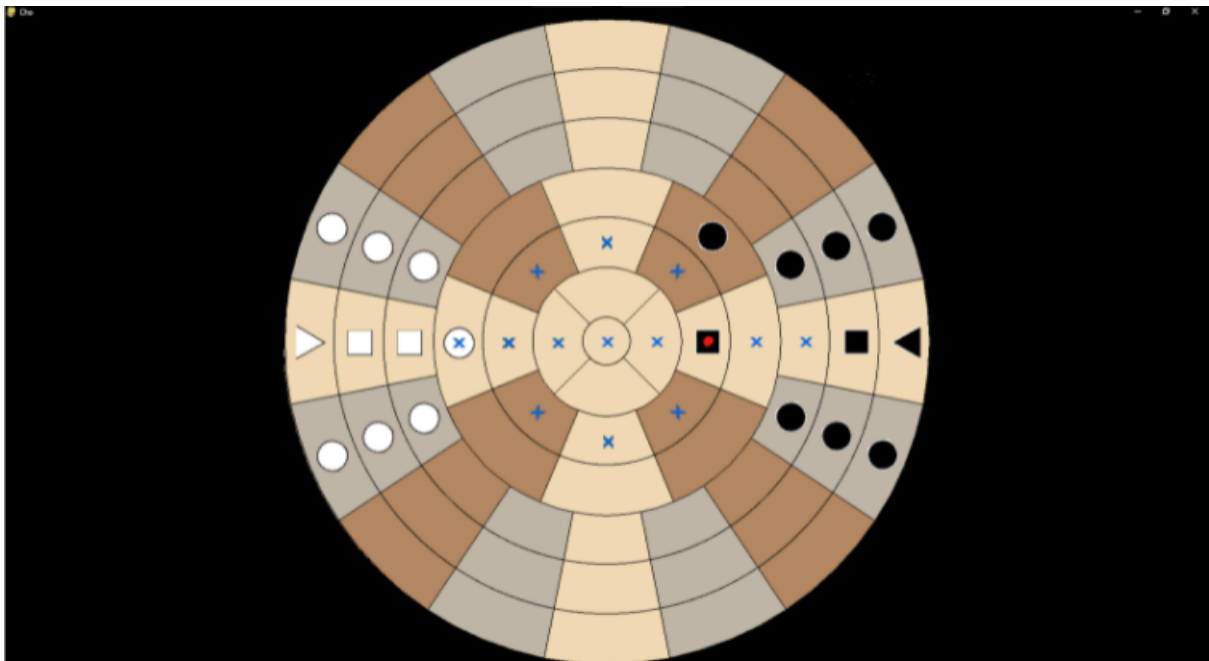


Figura 7.1: Referenciando as casas possíveis para a movimentação da Torre destacada.

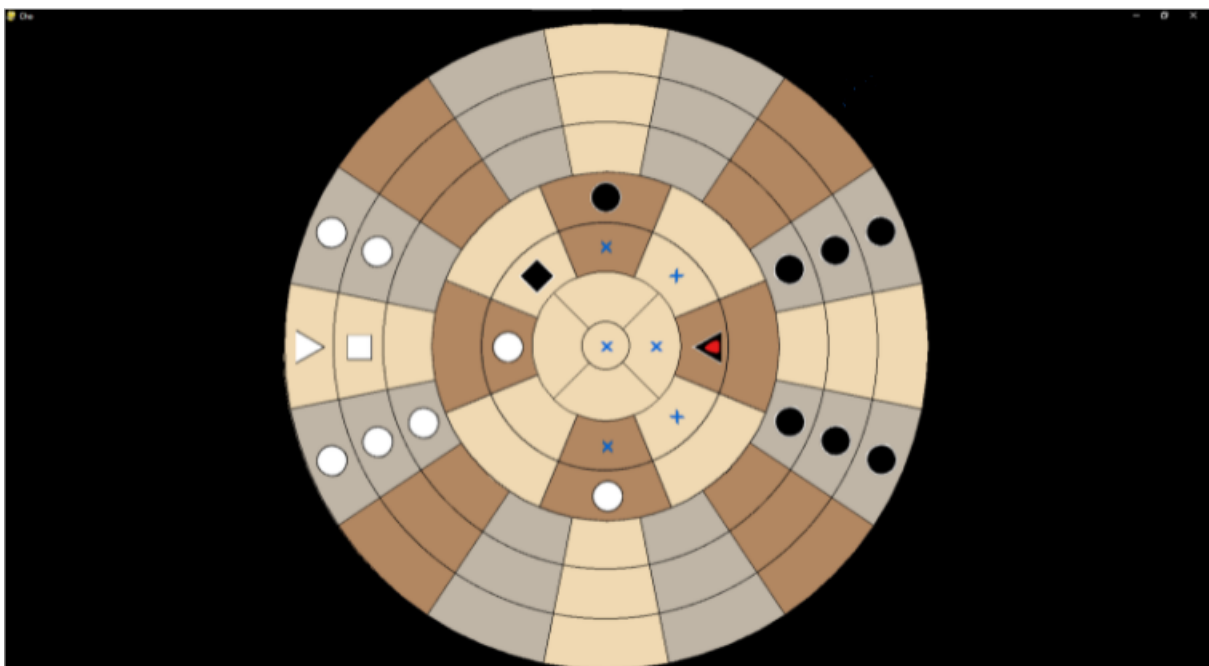


Figura 7.2: Referenciando as casas possíveis para a movimentação da Rainha destacada.



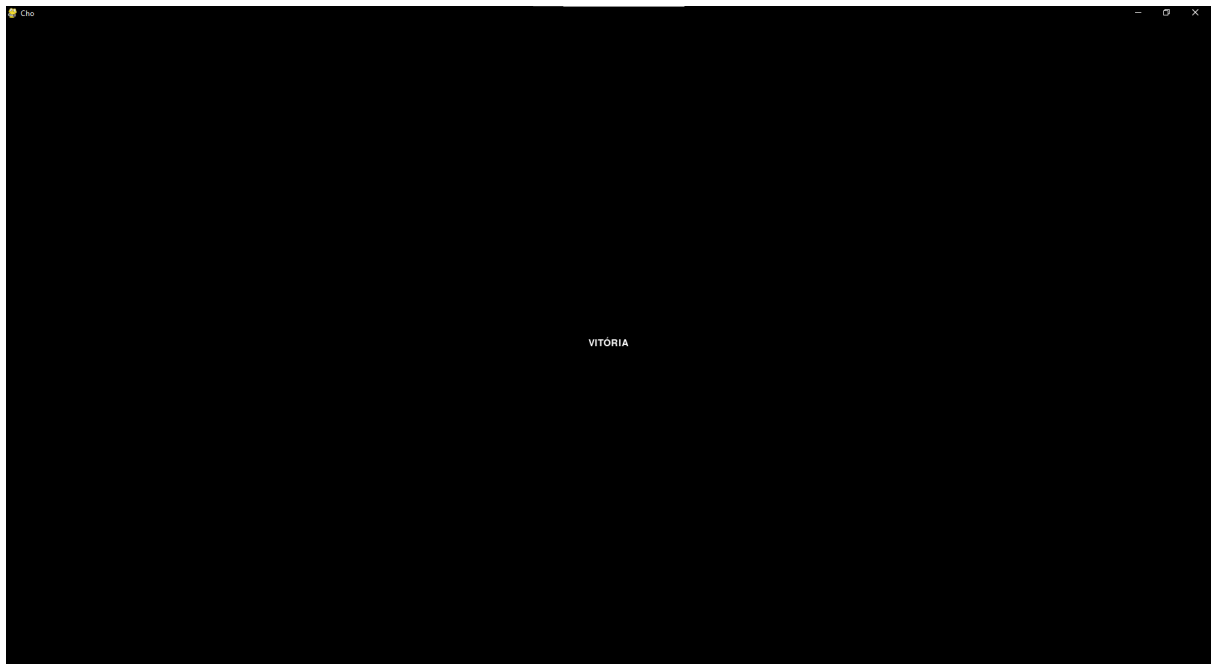


Figura 8: Tela de vitória para o jogador com as peças pretas.



Figura 8.1: Tela de vitória para o jogador com as peças brancas.

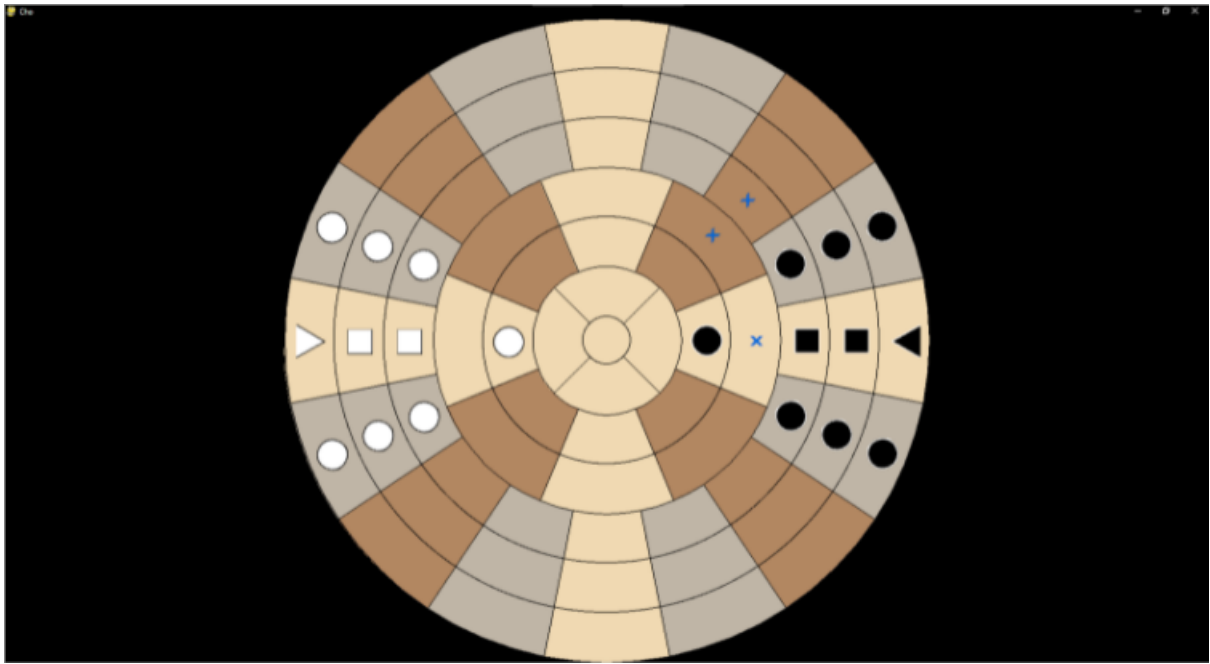


Figura 9: Interseção do peão.