

Cartões CRC

|   |  |
|---|--|
| <div><div>Class: Game</div><div>Responsability</div><div>Representa um jogo com estatísticas de desempenho. A classe Game armazena informações sobre um jogo específico, incluindo seu nome e estatísticas relacionadas ao desempenho, como o número de vitórias, derrotas e empates</div></div>  | <div><div>Class: Player</div><div>Responsability</div><div>Classe abstrata que define a estrutura e funcionalidades básicas do tabuleiro. A classe Player armazena informações como nome, apelido e estatísticas de desempenho em diferentes jogos, permitindo o gerenciamento e exibição dessas informações.</div></div> <div><div>Collaboration</div><div>Collors<br/>GameController</div></div>   |
| <div><div>Class: Colors</div><div>Responsability</div><div>Declaração de constantes para cores e estilos de texto para alterar a saída no terminal. Este arquivo define uma série de constantes para representar cores e estilos de texto que podem ser usados para colorir e formatar a saída do terminal. As cores são definidas usando códigos ANSI e são aplicadas a texto para melhorar a visibilidade e a estética.</div></div> | <div><div>Class: Board</div><div>Responsability</div><div>Classe abstrata que define a estrutura e funcionalidades básicas de um tabuleiro de jogo. A classe Board é uma classe abstrata que define as operações essenciais para manipular um tabuleiro de jogo. As classes derivadas devem implementar o método virtual puro play para realizar jogadas específicas do jogo.</div></div> <div><div>Collaboration</div><div>Cell<br/>Player<br/>Colors</div></div>   |
| <div><div>Class: Cell</div><div>Responsability</div><div>Representa uma célula individual em um tabuleiro de jogo. A classe Cell é usada para representar uma célula no tabuleiro. Ela armazena o valor da célula e um estado booleano que indica se a célula foi selecionada ou não.</div></div>   | <div><div>Class: Interface</div><div>Responsability</div><div>Gerencia a interação com o usuário através de menus e operações relacionadas a jogadores e jogos. A classe Interface fornece métodos para exibir menus, criar e remover jogadores, realizar login de jogadores, e gerenciar a interação geral com o usuário. Ela se comunica com outras classes, como GameController, para manipular dados dos jogadores e iniciar diferentes jogos.</div></div> <div><div>Collaboration</div><div>Player,<br/>ControllerMinesweeper,<br/>Reversi, Lig4,<br/>TicTacToe, Checkers</div></div> |
|   | <div><div>Class: Checkers</div><div>Responsability</div><div>Implementa a lógica do jogo de damas (checkers) baseado na classe Board. A classe CheckersGame é uma implementação específica de um jogo de damas que herda da classe Board. Ela inclui métodos para definir o tabuleiro, validar movimentos, realizar jogadas e determinar o vencedor do jogo.</div></div> <div><div>Collaboration</div><div>Board</div></div>   |
| <div><div>Class: GameController</div><div>Responsability</div><div>Gerencia a lista de jogos disponíveis no projeto. A classe GameController é responsável por manter uma lista de jogos disponíveis e fornecer acesso a esses jogos. A lista de jogos é gerenciada internamente.</div></div>   | <div><div>Collaboration</div><div>Game</div></div>   |
| <div><div>Class: Reversi</div><div>Responsability</div><div>Classe que implementa o jogo Reversi. A classe ReversiGame herda da classe Board e implementa as regras e a lógica específicas do jogo Reversi, também conhecido como Othello. Esta classe gerencia o tabuleiro, a troca de jogadores, a validação de movimentos, o cálculo da pontuação, e o resultado final do jogo.</div></div>  | <div><div>Collaboration</div><div>Board</div></div>  |
| <div><div>Class: DatabaseController</div><div>Responsability</div><div>Gerencia a leitura e escrita de dados de jogadores em arquivos. A classe DatabaseController é responsável por carregar e salvar informações sobre jogadores a partir e para arquivos. Ela utiliza os métodos de leitura e escrita para interagir com a classe GameController.</div></div>  | <div><div>Collaboration</div><div>PlayerController</div></div>   |
| <div><div>Class: Minesweeper</div><div>Responsability</div><div>Implementa o jogo Campo Minado. A classe Minesweeper herda da classe Board e fornece a lógica específica para o jogo Campo Minado, incluindo a colocação de bombas, verificação de derrota e vitória, e exibição do tabuleiro com informações sobre as células adjacentes.</div></div>  | <div><div>Collaboration</div><div>Board</div></div>  |
| <div><div>Class: PlayerController</div><div>Responsability</div><div>Controlador responsável por gerenciar operações relacionadas aos jogadores. A classe PlayerController lida com a adição, remoção, e acesso de jogadores dentro do sistema. Ela também fornece métodos para listar jogadores, tanto por nome quanto por apelido (nickname).</div></div>   | <div><div>Collaboration</div><div>Player<br/>DatabaseController</div></div>  |
| <div><div>Class: TicTacToe</div><div>Responsability</div><div>Declaração da classe TicTacToe para implementar o jogo da velha (Tic-Tac-Toe).</div></div>  | <div><div>Collaboration</div><div>Board</div></div>  |
| <div><div>Class: Lig4</div><div>Responsability</div><div>Declaração da classe Lig4 para implementar o jogo Lig4.</div></div>  | <div><div>Collaboration</div><div>Board</div></div>  |