

## G01

(28%) Francisco Magalhães - up202005141

(44%) Miguel Borges - up202004481

(28%) Tomás Cruz - up202008274

### TEMPO DE DESENVOLVIMENTO

Total: 53 horas

### ESPECIFICAÇÕES ADICIONAIS

Para além das classes mencionadas no guião (Player e Board), foram criadas:

BoardState, Computer (derivada de Player), Game e uma template class InputHandler.

Estas são descritas em mais detalhe na seguinte documentação, criada pelo grupo:

[Connect 4 - EDA Project 2: Game Class Reference \(miguelpcborges.github.io\)](https://miguelpcborges.github.io/Connect4-EDA-Project2-Game-Class-Reference/)

### ESTADO DE DESENVOLVIMENTO

Todos os objetivos para o projeto foram concluídos com sucesso.

### PRINCIPAIS DIFICULDADES

- A programação da AI foi talvez o mais desafiante, uma vez que a implementação do algoritmo de MinMax implicou a simulação de vários Boards, cuja quantidade crescia exponencialmente e cuja memória tinha de ser alocada e apagada dinamicamente.
- Foi necessária a dedicação de uma quantidade considerável de tempo para debug, uma vez que a implementação de classes template tem nuances que, a princípio, não eram totalmente compreendidas pelo grupo.
- Uma vez que foi criada uma classe Computer, derivada de Player, a identificação de cada player como sendo AI ou controlado por um jogador demonstrou-se desafiante, tendo sido resolvida com recurso a pointers e dynamic\_cast, como pode ser visto no destructor da classe Game.
- Caching da AI para otimizar a maneira como eram gerados os board states futuros necessitou uma reestruturação completa da maneira como estes estavam a ser organizados, implicando selecionar a memória correspondente ao board state desejado, sendo que esta nem sempre estava disponível.