|  |
| --- |
| **IArtificialIntelligence**  **<<Interface>>** |
| + Hit() : Field  + GetDifficulty() : Difficulty |

|  |
| --- |
| **EasyAI** |
| - \_userBoard : Board  - \_fieldList : List<Field>  - \_notHitFieldsList : List<Field> |
| + EasyAI(Board userBoard)  + GetDifficulty() : Difficulty  + Hit(): Field |

|  |
| --- |
| **MediumAI** |
| - \_userBoard : Board  - \_easyAI : IArtificialIntelligence  - \_hardAI : IArtificialIntelligence  - \_fieldList : List<Field>  - \_difficltyToUse : List<Difficulty> |
| + MediumAI(Board userBoard)  + GetDifficulty() : Difficulty  + Hit() : Field |

|  |
| --- |
| **HardAI** |
| - \_userBoard : Board  - \_fieldList : List<Field>  - \_hitShipList : List<Field>  - \_hitWaterList : List<Field>  - \_destroyedShipsList : List<Ship>  - \_mayHaveShipsAnySize : List<Field>  - \_mayHaveShipsSize2 : List<Field>  - \_mayHaveShipsSize3 : List<Field>  - \_mayHaveShipsSize4 : List<Field>  - \_mayHaveShipsSize5 : List<Field>  - \_mayHaveRestOfShip : List<Field> |
| + HardAI(Board userBoard)  + Hit() : Field  + GetDifficulty() : Difficulty  - GetSurroundingFields(Field field) : List<Field>  - GetLeftField(Field field) : Field  - GetUpField(Field field) : Field  - GetRightField(Field field) : Field  - GetDownField(Field field) : Field  - GetMayHaveShips(int size) : List<Field>  - GetLargestPossibleShipFileds(Field field, Direction direction) : List<Field> |

|  |
| --- |
| **Cooordinate**  **<<Struct>>** |
| + x : get; set  + y : get; set |
| + Coordinate(int x, int y) |

|  |
| --- |
| **Board** |
| - \_fields : Field[,]  - \_width : int  - \_height : int |
| + Board(int width, int height, bool isRevealed)  + Width : int  + Height : int  + GetField(Coordinate coordinate) : Field  + GetCoordinates(Field field) : Coordinate  - InitializeFields(bool isRevealed) : void |

|  |
| --- |
| **Field** |
| - \_isRevealed : bool  - \_ship : Ship  - \_shipFieldPosition : int  - \_shipDirection : Direction  - \_isHit : bool |
| + Field(Field copy)  + IsRevealed : bool  + IsHit : bool  + Ship : Ship  + ShipFieldPosition : int  + ShipDirection : Direction |

|  |
| --- |
| **Score** |
| - \_playerName : string  - \_easyWins : int  - \_mediumWins : int  - \_hardWins : int  - \_losses : int |
| + Score(string playerName, int easyWins, int mediumWins, int hardWins, int losses)  + PlayerName : string  + EasyWins : int  + MediumWins : int  + HardWins : int  + Losses : int |

|  |
| --- |
| **Ship** |
| - \_size : int  - \_actualSize : int  - \_isSunk : bool  - \_name : String |
| + Ship(int size)  + Ship(Ship copy)  + Size : int  + ActualSize : int  + IsSunk : bool  + Name : String  - ShipName(int size) : String |

|  |
| --- |
| **Difficulty**  **<<Enum>>** |
| + Easy  + Medium  + Hard |

|  |
| --- |
| **Direction**  **<<Enum>>** |
| + Horizontal  + Vertical |

|  |
| --- |
| **NoShipSpaceException**  **<<Exception>>** |
| + NoShipSpaceException() |

|  |
| --- |
| **ShipFactory** |
|  |
| + FillBoardRandomly(Board board, int numberOfSize5, int numberOfSize4, int numberOfSize3, int numberOfSize2) : void  + CreateShips(int numberOfSize5, int numberOfSize4, int numberOfSize3, int numberOfSize2) : List<Ship>  + VerifyField(Board board, Coordinate coordinate) : bool  - PlaceShip(Board board, Ship ship, Random rand) : void  - VerifyShipPlacement(Board board, Coordinate coordinate, Ship ship, Direction direction) : bool  - FindPossiblePlacements(Board board, Ship ship, Direction direction) : List<Coordinate> |

|  |
| --- |
| **Turn**  **<<Enum>>** |
| + Player  + Computer |