Board - board: List<List<Tile>> - pane: StackPane - display: Display - isEndOfGame: boolean - initializeBoard(): List<List<Tile>> - endGame(boolean): void - endGame(OOrX): void - initializeTile(x: int, y: int): Tile + getBoard(): List<List<Tile>> + startNewGame(): void + checkForWinner(): void + equals(Object): boolean + hashCode(): int + toString(): String 1 contains * Tile - type: OOrX - pane: StackPane - label: Label - board: Board - resetType(): void - play(): void + setTile(OOrX): void + resetTile(): void + getType(): OOrX + equals(Object): boolean + hashcode(): int + toString(): String is of type <<enumeration>> OOrX 0 Χ

Calculation

- SIZE: int
- MAX_SCORE: int
- MIN_SCORE: int
- + minimax(Board): List<Integer>
- + calculateRow(Board): OOrX
- + calculateColumn(Board): OOrX+ calculateDiagonal(Board): OOrX
- + calculateStalemate(Board): boolean