«GameObject» nl.tudelft.ti2206.gameobjects::Grid -logger = Logger.getInstance(): Logger -className = this.getClass().getSimpleName(): String -GRID WIDTH = 400: int -GRID HEIGHT = 400: int -GRID X = 100: int -GRID Y = 100: int -NTILES = 16: int -TWO = 2: int -FOUR = 4: int -region: TextureRegion -tiles: Tile[] -random: Random -iterator: TileIterator -tileHandler: TileHandler -highestTile: int -score: int -highScore: int +Grid(boolean isEmpty): ctor +Grid(boolean isEmpty, Skin skin, TextureRegion texture): ctor +setTile(int index. int value): void +act(float delta): void +restart(): void +move(Direction direction): void +updateHighestTile(): void +getPossibleMoves(): int +getTileNeighbors(int index): List<Tile> +getTiles(): Tile[] +aetCurrentHighestTile(): int +getScore(): int +getHighscore(): int +getTileHandler(): TileHandler +setTileHandler(TileHandler tileHandler): void +setHighestTile(int highest): void +setScore(int score): void +setHighscore(int highScore): void +getX(): float +getY(): float +qetWidth(): float

+getHeight(): float

+toString(): String

+draw(Batch batch, float parentAlpha): void

«Handler» nl.tudelft.ti2206.handlers::TileHandler -ROW LENGTH = 4: int -tiles: Tile[] -isMoveMade: boolean -scoreIncrement: int +TileHandler(Tile[] tiles): ctor +moveLeft(): void uses +moveRight(): void +moveDown(): void +moveUp(): void +isMoveMade(): boolean +getScoreIncrement(): int +reset(): void 1 uses ùses «GameObject» 1 nl.tudelft.ti2206.gameobjects::Tilelterato -tiles: Tile[] -index: int +TileIterator(Tile[] tiles): ctor +hasNext(): boolean +next(): Tile +reset(): void +getIndex(): int

+remove(): void