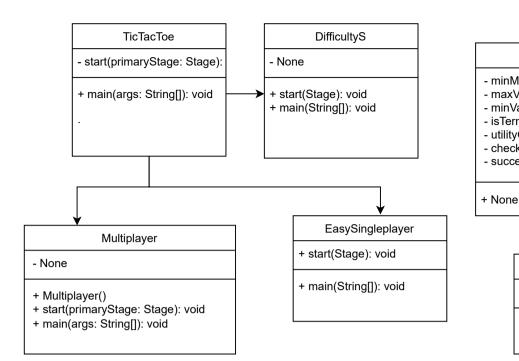
# Tic Tac Toe Game



# AdversarialSearchTicTacToe - minMaxDecision(state: State): int - maxValue(state: State): int - minValue(state: State): int - isTerminal(state: State): boolean - utilityOf(state: State): int - checkState(state: State, a: int): String - successorsOf(state: State): ArrayList<State>

# HardSingleplayer

+ start(Stage): void

+ main(String[]): void

#### ControllerHardS

- random: Random
- buttons: ArrayList<Button>
- ticTacToeAl: AdversarialSearchTicTacTc
- button1: Button
- button2: Button
- button3: Button
- button4: Button
- button5: Button
- button6: Button
- button7: Button
- button8: Button
- button9: Button
- winnerText: Text
- + initialize(url: URL,
- resourceBundle:ResourceBundle): void
- + restartGame(event: ActionEvent): void + exitGame(event: ActionEvent): void
- + resetButton(button: Button): void
- + exitButton(button: Button): void
- + setupButton(button: Button): void
- + makeAlMove(): void
- + pickButton(index: int): void
- + getBoardState(): State
- + checklfGameIsOver(): void

### ControllerEasyS

- random: Random
- buttons: ArrayList
- button1: Button
- button2: Button
- button3: Button
- button4: Button
- button5: Button
- button6: Button
- button7: Button
- button8: Button
- button9: Button
- winnerText: Text
- + initialize(URL, ResourceBundle): void
- + restartGame(ActionEvent): void
- + exitGame(ActionEvent): void
- + resetButton(Button): void
- + exitButton(Button): void
- + setupButton(Button): void
- + makeAlMove(): void + pickButton(int): void
- + getBoardState(): State
- + checkIfGameIsOver(): void

## State

- position: int
- state: String[]
- + State(position: int, state: String[])
- + getPosition(): int
- + setPosition(position: int): void
- + getState(): String[]
- + getStateIndex(i: int): String
- + setState(state: String[]): void
- + changeState(i: int, player: String): void
- + toString(): String

### ControllerM

- random: Random
- buttons: ArrayList<Button>
- isPlayerOneTurn: boolean
- + initialize(URL, ResourceBundle): void
- + resetButton(Button): void
- + checklfGameIsOver(): void
- + exitButton(Button): void
- + pickButton(int): void
- + getBoardState(): String[]
- + setupButton(Button): void