## Controller Package

+ SelectValue extends EffectGame

+ eventViewToAsk (): EventClient

trueSetValueFalseIncrementDec : boolean

SelectValue (trueSetValueFalseIncrementDec: boolean)

+ doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void

valueDice:int

+ undo():void

constructors-

+ EndTurn extends EffectGame

+ EndTurn(special: boolean)

+ eventViewToAsk ():EventClient

+ doEffect(gameBoard:GameBoard, idPlayer:int, infoMove:int[]):void

special:boolean

-constructors -

+ undo():void

implements ControllerVisitor TimerCallback ∃fields —— gameBoard:GameBoard playerNumber:int restoreAble:boolean server:ServerController gameRoom: GameRoom effectToRead : LinkedList < EffectGame > currentEffect:int effectGamesStored : LinkedList < EffectGame > PLAYER\_TIMEOUT:long playerTimeout:TimerThread updaterView : UpdaterView - started:boolean + Controller(server:ServerController, playerName:String[], room:GameRoom) + getUpdater():UpdateRequestedByServer + startController():void + endGame():void playerDown(index:int):void + winBecauseOfDisconnection (winnerld:int):void playerUp(index:int):void sendInitCommand():void + sendEventToController(event: EventController):void sendEventToView(event: EventClient):void + visit(event: ControllerSelectInitialWindowPatternCard ):void + visit(event: ControllerMoveDrawAndPlaceDie ):void + visit(event: ControllerMoveUseToolCard):void + synchronized visit(event: ControllerEndTurn):void + visit(event: ControllerSelectDiceFromDraftPool):void + visit(event: ControllerSelectCellOfWindow):void visit(event: ControllerSelectToolCard):void + visit(event: ControllerInfoEffect):void - accessToEffect (idPlayer: int, info:int[]):void showErrorMessage(ex:Exception, idPlayer:int, showMenuTurn:boolean):void sendWaitTurnToAllTheNonCurrent(currentPlayerId: int):void + timerCallback ():void

+ timerCallbackWithIndex (index:int):void

InsertDice extends EffectGame line:int column:int adjacentR:boolean colorR:boolean valueR:boolean firstDieOfTheTurn:boolean InsertDice (adjacentR: boolean, colorR: boolean, valueR: boolean, firstDieOfTheTurn: boolean) doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void + undo():void + eventViewToAsk ():EventClient

+ undo():void + eventViewToAsk ():EventClient getGameBoard (): GameBoard getIdPlayer ():int setGameBoard (gameBoard: GameBoard):void setIdPlayer (idPlayer: int):void FactoryEffect extends EffectGame + ChangeDiceValue extends EffectGame trueRandomFalseOpposite:boolean doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void ChangeDiceValue (trueRandomFalseOpposite: boolean) eventViewToAsk ():EventClient doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void undo():void + eventViewToAsk (): EventClient

doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void

+ EffectGame

idPlayer:int

→ methods —

implements Serializable

gameBoard:GameBoard

+ RemoveDiceFromWindow extends EffectGame line:int column:int

trueSpecialRemoveFalseNormal:boolean constructors -

RemoveDiceFromWindow(trueSpecialRemoveFalseNormal: boolean) ---methods -----

doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void + undo():void undo():void

round:int

index:int

trueSwapDiceFalseSetColorRestriction:boolean

RoundTrackEffect(trueSwapDiceFalseSetColorRestriction: boolean)

eventViewToAsk ():EventClient + eventViewToAsk ():EventClient

+ RoundTrackEffect extends EffectGame + DicePoolEffect extends EffectGame

indexDiceOfDicePool:int

trueDrawDieFalseRollDice:boolean

+ DicePoolEffect (trueDrawDieFalseRollDice: boolean)

+ doEffect(gameBoard: GameBoard, idPlayer: int, infoMove: int[]):void

+ undo():void

+ eventViewToAsk (): EventClient