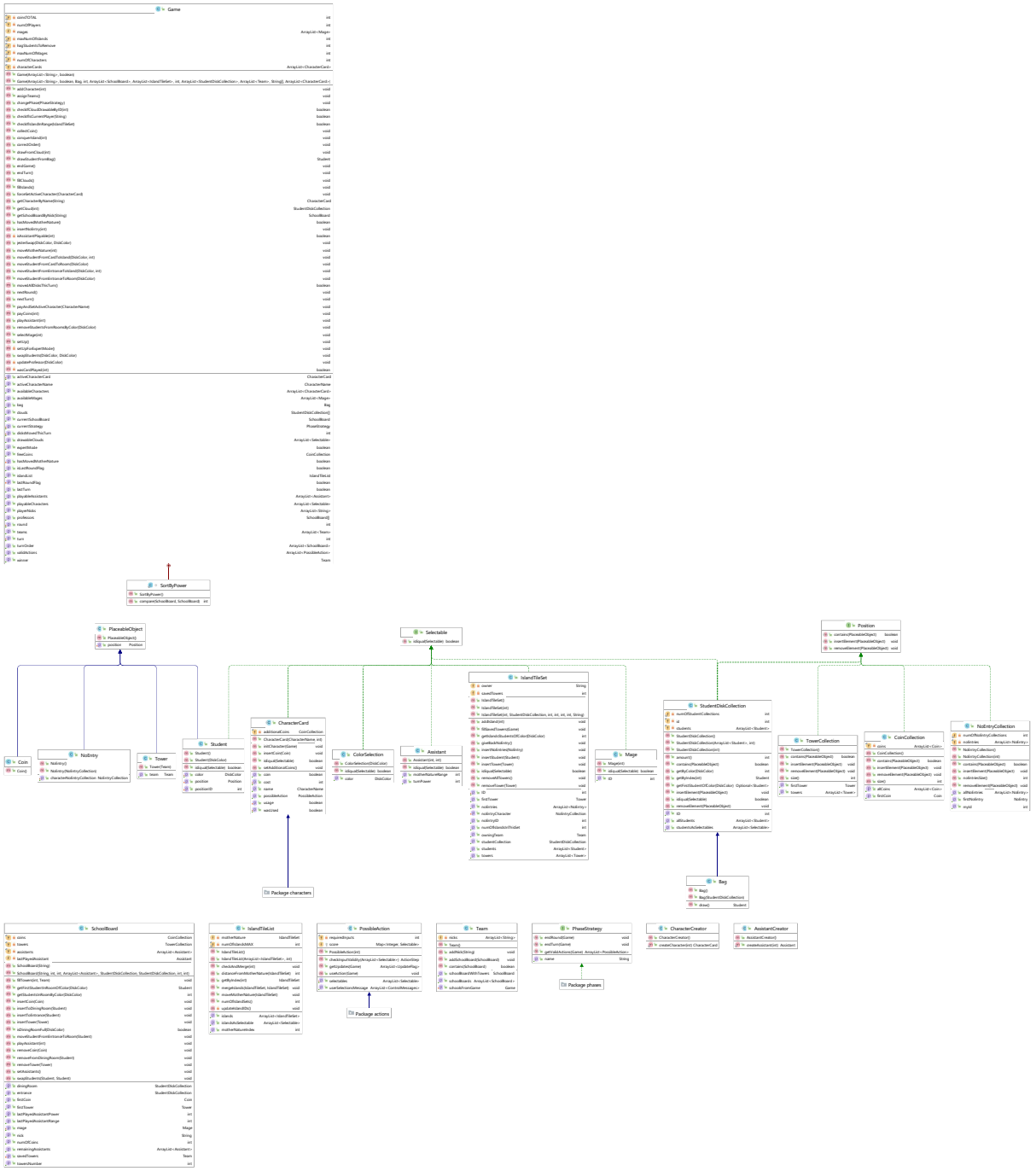
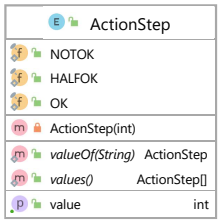
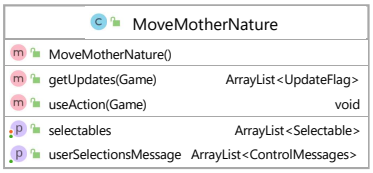
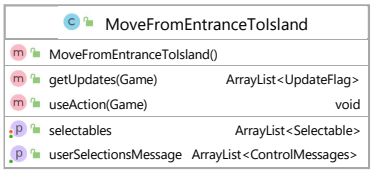
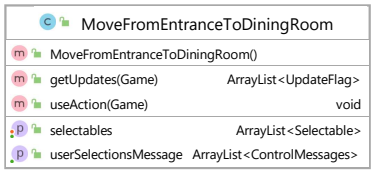
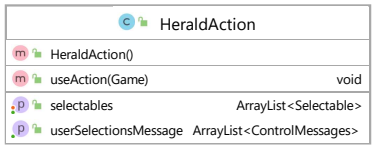

Game
SortByPower
PlaceableObject
Selectable
Position
Coin
NoEntry
Tower
Student
CharacterCard
ColorSelection
Assistant
IslandTileSet
Mage
StudentDiskCollection
TowerCollection
CoinCollection
NoEntryCollection
Package characters
Bag
SchoolBoard
IslandTileList
PossibleAction
Team
PhaseStrategy
CharacterCreator
AssistantCreator
Package actions
Package phases

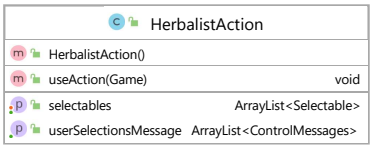

ActionStep
NOTOK
HALFOK
OK
ActionStep(int)
valueOf(String) ActionStep
values() ActionStep[]
value int

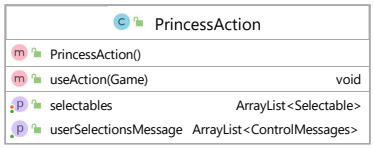

MoveMotherNature
MoveMotherNature()
getUpdates(Game) ArrayList<UpdateFlag>
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

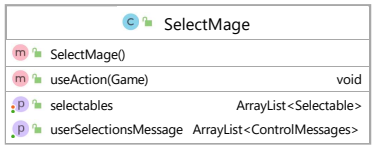

MoveFromEntranceToIsland
MoveFromEntranceToIsland()
getUpdates(Game) ArrayList<UpdateFlag>
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

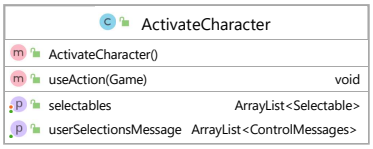

MoveFromEntranceToDiningRoom
MoveFromEntranceToDiningRoom()
getUpdates(Game) ArrayList<UpdateFlag>
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

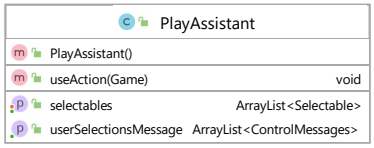

HeraldAction
HeraldAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

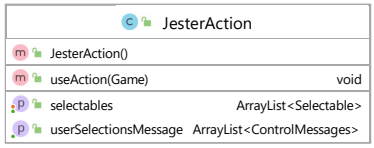

HerbalistAction
HerbalistAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

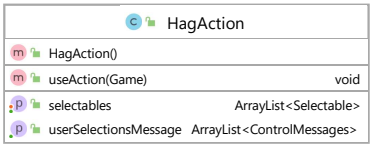

PrincessAction
PrincessAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

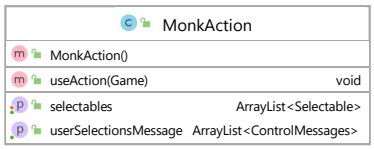

SelectMage
SelectMage()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

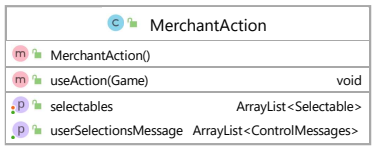

ActivateCharacter
ActivateCharacter()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

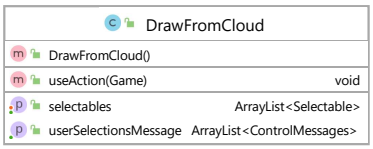

PlayAssistant
PlayAssistant()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

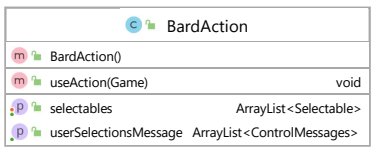

JesterAction
JesterAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

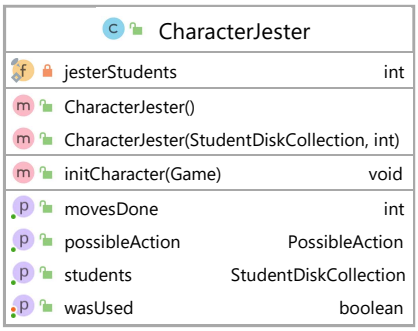

HagAction
HagAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

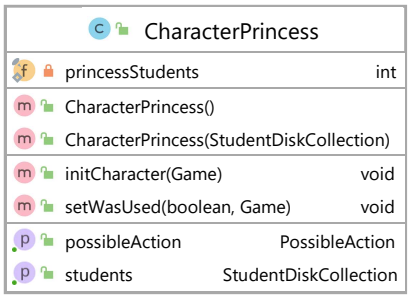

MonkAction
MonkAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

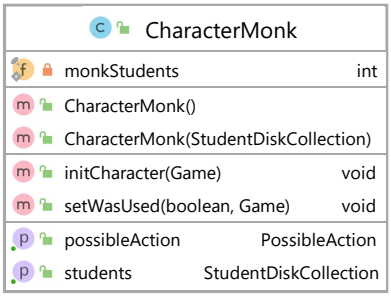

MerchantAction
MerchantAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

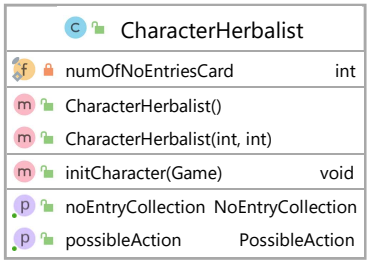

DrawFromCloud
DrawFromCloud()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

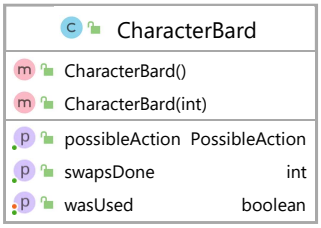

BardAction
BardAction()
useAction(Game) void
selectables ArrayList<Selectable>
userSelectionsMessage ArrayList<ControlMessages>

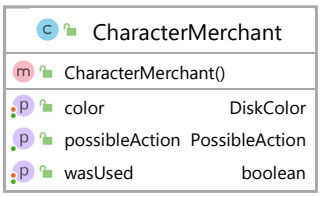

CharacterJester
jesterStudents int
CharacterJester()
CharacterJester(StudentDiskCollection, int)
initCharacter(Game) void
movesDone int
possibleAction PossibleAction
students StudentDiskCollection
wasUsed boolean

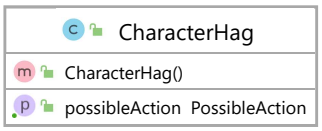

CharacterPrincess
princessStudents int
CharacterPrincess()
CharacterPrincess(StudentDiskCollection)
initCharacter(Game) void
setWasUsed(boolean, Game) void
possibleAction PossibleAction
students StudentDiskCollection

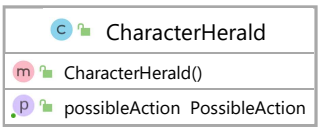

CharacterMonk
monkStudents int
CharacterMonk()
CharacterMonk(StudentDiskCollection)
initCharacter(Game) void
setWasUsed(boolean, Game) void
possibleAction PossibleAction
students StudentDiskCollection

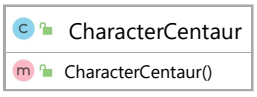

CharacterHerbalist
numOfNoEntriesCard int
CharacterHerbalist()
CharacterHerbalist(int, int)
initCharacter(Game) void
noEntryCollection NoEntryCollection
possibleAction PossibleAction

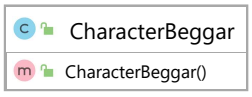

CharacterBard
CharacterBard()
CharacterBard(int)
possibleAction PossibleAction
swapsDone int
wasUsed boolean

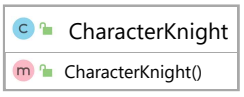

CharacterMerchant
CharacterMerchant()
color DiskColor
possibleAction PossibleAction
wasUsed boolean

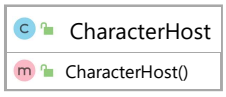

CharacterHag
CharacterHag()
possibleAction PossibleAction


CharacterHerald
CharacterHerald()
possibleAction PossibleAction


CharacterCentaur
CharacterCentaur()


CharacterBeggar
CharacterBeggar()


CharacterKnight
CharacterKnight()


CharacterHost
CharacterHost()

SetupStrategy
SetupStrategy()
endRound(Game) void
endTurn(Game) void
getValidActions(Game) ArrayList<PossibleAction>
name String
ActionStrategy
ActionStrategy()
endRound(Game) void
endTurn(Game) void
getValidActions(Game) ArrayList<PossibleAction>
name String
PlanningStrategy
PlanningStrategy()
endRound(Game) void
endTurn(Game) void
getValidActions(Game) ArrayList<PossibleAction>
name String