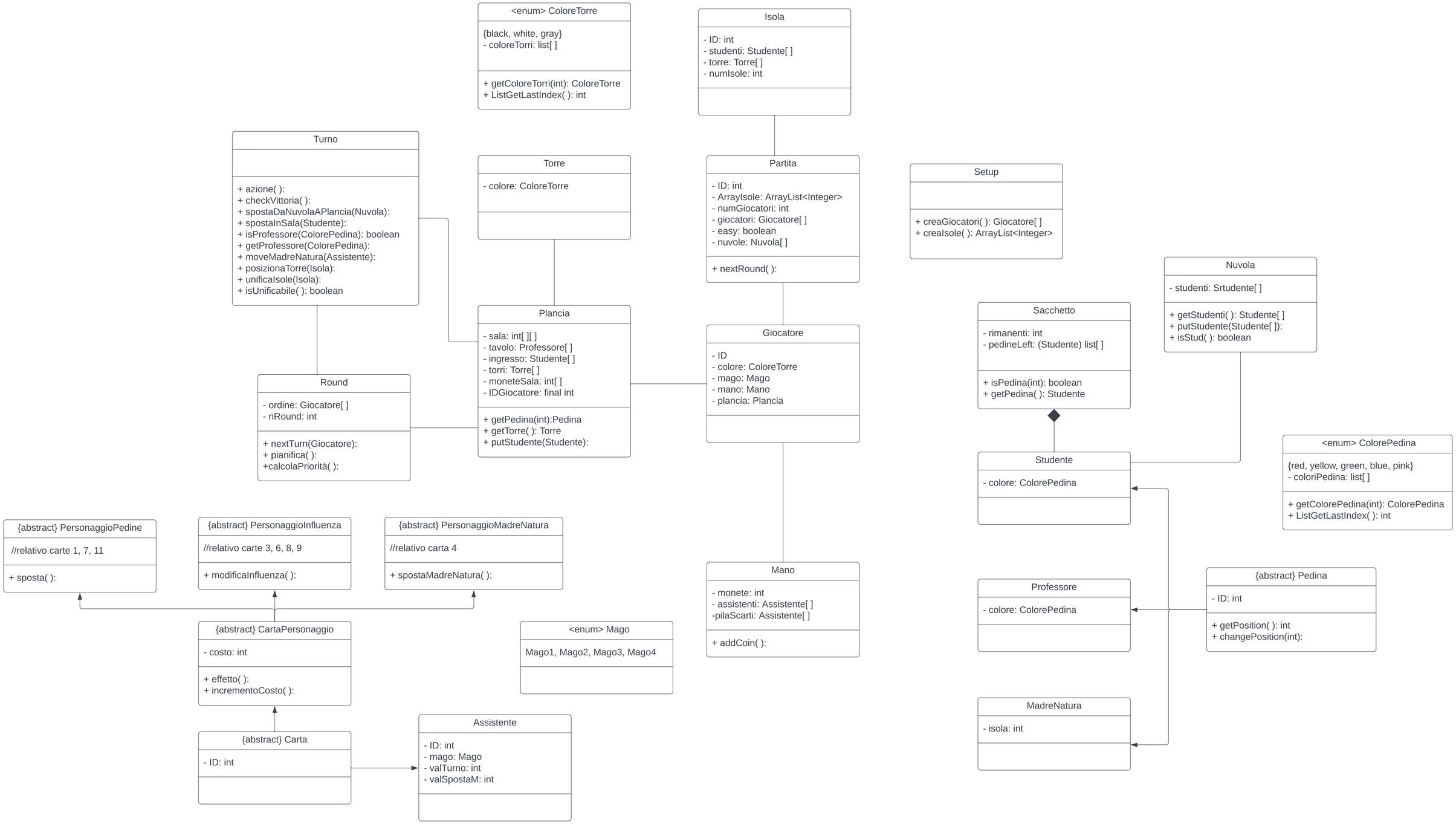
<enum> ColoreTorre
{black, white, gray}
- coloreTorri: list[ ]
+ getColoreTorri(int): ColoreTorre
+ ListGetLastIndex( ): int
Isola
- ID: int
- studenti: Studente[ ]
- torre: Torre[ ]
- numIsole: int
Turno
+ azione( ):
+ checkVittoria( ):
+ spostaDaNuvolaAPlancia(Nuvola):
+ spostaInSala(Studente):
+ isProfessore(ColorePedina): boolean
+ getProfessore(ColorePedina):
+ moveMadreNatura(Assistente):
+ posizionaTorre(Isola):
+ unificaIsole(Isola):
+ isUnificabile( ): boolean
Torre
- colore: ColoreTorre
Partita
- ID: int
- ArrayIsole: ArrayList<Integer>
- numGiocatori: int
- giocatori: Giocatore[ ]
- easy: boolean
- nuvole: Nuvola[ ]
+ nextRound( ):
Setup
+ creaGiocatori( ): Giocatore[ ]
+ creaIsole( ): ArrayList<Integer>
Nuvola
- studenti: Srtudente[ ]
+ getStudenti( ): Studente[ ]
+ putStudente(Studente[ ]):
+ isStud( ): boolean
Plancia
- sala: int[ ][ ]
- tavolo: Professore[ ]
- ingresso: Studente[ ]
- torri: Torre[ ]
- moneteSala: int[ ]
- IDGiocatore: final int
+ getPedina(int):Pedina
+ getTorre( ): Torre
+ putStudente(Studente):
Giocatore
- ID
- colore: ColoreTorre
- mago: Mago
- mano: Mano
- plancia: Plancia
Sacchetto
- rimanenti: int
- pedineLeft: (Studente) list[ ]
+ isPedina(int): boolean
+ getPedina( ): Studente
Round
- ordine: Giocatore[ ]
- nRound: int
+ nextTurn(Giocatore):
+ pianifica( ):
+calcolaPriorità( ):
Studente
- colore: ColorePedina
<enum> ColorePedina
{red, yellow, green, blue, pink}
- coloriPedina: list[ ]
+ getColorePedina(int): ColorePedina
+ ListGetLastIndex( ): int
{abstract} PersonaggioPedine
//relativo carte 1, 7, 11
+ sposta( ):
{abstract} PersonaggioInfluenza
//relativo carte 3, 6, 8, 9
+ modificaInfluenza( ):
{abstract} PersonaggioMadreNatura
//relativo carta 4
+ spostaMadreNatura( ):
Mano
- monete: int
- assistenti: Assistente[ ]
-pilaScarti: Assistente[ ]
+ addCoin( ):
Professore
- colore: ColorePedina
{abstract} Pedina
- ID: int
+ getPosition( ): int
+ changePosition(int):
{abstract} CartaPersonaggio
- costo: int
+ effetto( ):
+ incrementoCosto( ):
<enum> Mago
Mago1, Mago2, Mago3, Mago4
MadreNatura
- isola: int
{abstract} Carta
- ID: int
Assistente
- ID: int
- mago: Mago
- valTurno: int
- valSpostaM: int