UML model diagram Eryantis - Write-up (Group GC33)

The following UML diagram represents the model of the MVC design pattern for the board game Eryantis.

It contains several classes defined as follows:
-A **Game** class with access to every other part of the model, even if not directly
-A **Player** class that defines a user of the program who is actively in a Game
-Two enum classes that store the possible student **Colour** and player **Team** values, with various methods to generate collections of that data type (e.g. Maps for Tile objects)
-An abstract **Tile** class (implemented by SchoolBoard, Bag, Cloud and Island) that represents an object capable of storing an EnumMap named contents that maps the Colour enum to an Integer, thus storing the amount of students per colour, and all methods associated with handling that attribute
-A **SchoolBoard** class with various methods and attributes that correspond to the individual player's board state and possible moves, the inherited contents attribute defines its entrance
-A **Bag** class with the added method of being able to draw from its contents randomly
-A **Cloud** class which allows itself to be filled or emptied completely (i.e. from a Bag object to a player's SchoolBoard entrance)
-An **Island** class, defining a Tile which can be conquered by the players by gaining influence, since the amount of them shrinks during a Game and the order must not change, they are linked by referencing one another
-An **AssistantCard** class, which describes cards that are played every turn by the players and has a method that allows their initialisation from pre-determined values located in a resource file
-A **CharacterCard** class, which has the same generation method that AssistantCard has, but with different values and attributes. They are only initialised in the event of an expert Game