System Design

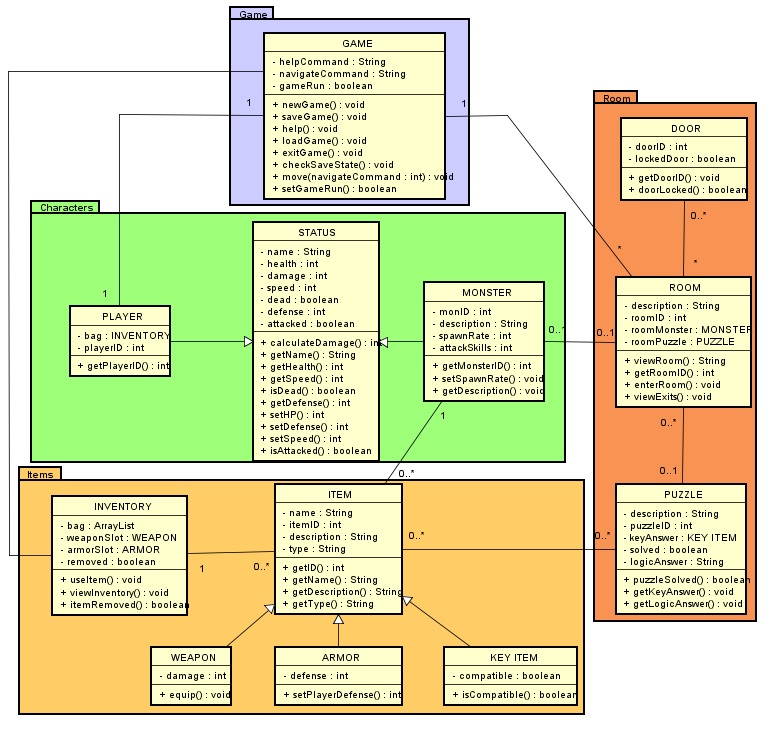
Team Cobra

**DESIGN CONSTRAINTS:**

This section will contain a list of nonfunctional requirements for the text-based adventure game. Also discuss any trades-off between nonfunctional requirements you had to make.

* System should respond to user input within ½ second.
* The different features of the puzzle window should be clear and easy to use. Text parser should always be available. Journal should be directly opened and user should not have to open inventory first. Puzzle window should close automatically after user enters an answer.
* The system must store the data for only one saved game at a time. If a “new game” is started with a current “saved game” stored, then the current “saved game” data will remain until the user attempts to save the “new game”, in which case the “new game”’s data will overwrite the existing “saved game”’s data.
* Reliability that the system provides a response to user dialog. Provide feedback on 100% of user input.
* Internet is only needed to download the app and install any updates.
* Admin should have a specific way to log in to game. Prevent anyone who does not have administrative rights from editing game data such as monster and puzzle locations or HP for user or monsters.
* Test environments should be built for the application to allow testing of functions such as monster battles and puzzles.
* The software will be using text files to input and output data. Data containing monster, artifact, and puzzle assets can only be read, while saved data that are stored is able to read and write.

**SYSTEM DIAGRAM:**



**SUBSYSTEM DESCRIPTION:**

**Game Subsystem:**

* Description: This subsystem contains only one class diagram and its main functionality is to track the player and room interaction in two subsystem: characters and room. Its purpose to track the state of all persistent objects within the game system then update its behavior change between the interactions between room and characters subsystems. New game initializes player object and room array object.

**Characters Subsystem:**

* Description: This subsystem contains three classes; Player, Monster, and Status. It is utilized during battles to keep track of attributes and stats of the player and monsters such as HP, attack, defense, and speed.

**Room Subsystem:**

* Description: This subsystem contains three classes. These classes are Door, Room, and Puzzle. The main function of the Room subsystem is to allow the user to navigate between rooms and gives information about the rooms, doors, and puzzles.

**Items Subsystem:**

* Description: This subsystem contains five classes. Two main classes, Inventory and Item, and three subclasses of item for usable items. The main function of the Items subsystem is to provide a way to store and use items that can be found throughout the game. This subsystem will give the ability to store, view, and use items depending upon the nature of the item.