Actions

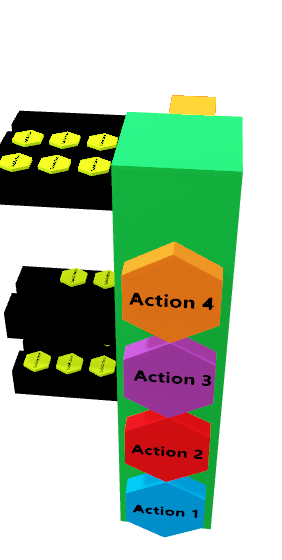
General Overview

We define as Action every movement or task that a **TDLCharacter** can perform (e.g. walk to reach a precise point, examine an object, talk to a different character, …). The ActionSystem aims to achieve mainly **two** tasks:

1. The sequential execution and queue of Actions for a single TDLCharacter, forcing the execution of an Action, eliminating the queue of precedent Actions, if has a major *priority*.
2. The interconnection between different elements of the game, permitting to different TDLCharacters (and other elements too) to communicate and dynamically perform their Actions in a *sync* or *async* way (e.g. during a cutscene, after a character has performed an Action, a different character should perform an Action itself).

Point 1 will be managed inside every TDLCharacter by a component called **ActionComponent**, which won’t communicate with any other element (excepted the ActionManager).

Point 2 will be managed globally in the current Room by a TDLRoomPlugin called **ActionManager**: its task consists in the communication between different ActionComponents.

ActionComponent

NOTA: ogni azione dovrà essere in lettura di input dall’esterno!

Ogni TDLCharacter avrà sempre un’azione che sta andando, se non sta facendo nulla sarà una IdleAction.