

New requirements for Sprint 3

Rough plan - implement RoomRules - e.g. the Snake Room and the Dragon Room have special rules. In this sprint we will also look at how an event can be notified to the GUI, e.g. if a key magically appears in a room, the GUI should be notified and updated.

Room rules

There are two rooms which are special in the game. In the database, they are listed as:

- ID 19
- ID 120

For room 19, in the old specification from the previous course, it was said that there is a Snake in the room, “blocking” the South exit. If we “put down” the Bird and the Cage, the Snake “Disappears” and the South exit will be set to the Room with ID 29 in the database.

For room 120, it was said that there is a Dragon here. If we “put down” Gold, Jewels, Diamonds and Silver, the Dragon “disappears” and a “Glass Key” appears in the Room. Also, the West exit is set to room id 0.

Our suggestion for room 120 it so skip the part of the West exit and focus on the appearing of the Glass Key instead.

How to manage database rooms

Now, we have a challenge with the fact that the Snake and Dragon are not in the database. So we need to “hard code” the rules for these rooms somehow. Luckily, there is a convenience method in the CaveInitializer class, for getting a reference to a Room via its database ID! This is implemented using the fact that all the rooms are actually stored in a `Map<Integer, Room>` where the keys are the database IDs and the values are the corresponding `se.itu.game.cave.Room:s`. This means that we can, as part of the initialization process (where stuff is read from the DB, `ThingRule:s` are added to the `RuleBook` etc) add rules for these two special rooms with the Snake and the Dragon.