Record and Play Functionality

Record and play functionality is achieved by storing the state of the board at the beginning of the recording and storing actions that took place during the recording. Using JSON frameworks and the javax.json-1.1 library, storing the game state and actions to a file was simplified.

I reused the save-to/load-from Json functionality written for level loading and saving to replicate the board state at the beginning of a recording. This method would save/load the state of the game including the time remaining, the board tiles, the player and their inventory and the state of all mobs.

Storing actions was done by recording the direction and agent for each event. All relevant events were movement-based (either player or mob) and thus all the needed to be stored was who moved, and where. Using the JSON framework, objects can be constructed to contain the required attributes such as the direction or agent, this allowed easy loading of this data when the file is parsed back into the program.

Having the state of the board and the list of actions, I allowed the user to step through the actions one at a time or play them back at a chosen speed. This functioned exactly as the regular gameplay does, with each action either calling movement on the player or a mob.

While running, all other gameplay systems had to be paused to prevent the user interfering or mob actions being called by the automatic timer. At the end of a recording, the time remaining had to be set to its correct value to prevent playing back a level on 10x speed for an advantage,

I disallowed recording to persist through death for simplicity's sake. This means that recordings in which the user dies are discarded and cannot be replayed.

Small quality of life additions includes a flag to indicate the system is replaying a level, disabling info field popups that would require user action to dismiss.

Zac Durant.