Project Work Structure

# Stage 1

* Create a document that fully details the different ghosts, items, rooms etc. so that it’s 100% clear what we’re implementing
* Create a file that holds a dictionary of ghosts (basically a file like items.py). The fields in the dictionary should be:
  + An ID for each ghost
  + The ghost’s name
  + A Description of the ghost for an examine function
  + An introductory speech given by the ghost as you enter combat with it for the first time
  + An alternate speech for when you enter combat after running from the ghost at least once
  + A Boolean field that shows whether the player has run from the ghost or not
  + The ghost’s combat skill (use explained below)
  + The ghost’s health
  + The number of conditions the player has met to peacefully deal with the ghost
* Update map.py so that rooms can have conditions that must be met before entering, and a variable that determines if a ghost is in a room or not.
* Assign the ghosts to the rooms in map.py
* Work on some means of determining the ‘state’ of the game; this could be used, for instance, to show an inventory menu, but is mainly used so that the normal main function does not keep displaying the room during combat and instead shows some kind of combat menu until combat ends.
* Get a basic placeholder combat scenario to work. The idea of how combat works is currently (this can be changed):
  + The player can fight the ghost. In this case random numbers determine a ‘dice roll’ for the player and the ghost, added to a combat skill value assigned to each. (If anyone is familiar with Fighting Fantasy choose-your-own-adventure books, it’s essentially that system) The person with the winning roll does a set number of damage to the loser. This way the player can get stronger through finding items and ghosts can have differing level of combat skill based on who they are.
  + The player can examine items, or the ghost. This works the same way as it does in rooms/inventory, as I implemented earlier
  + The player can use items: these may heal or be used to meet a condition for peacefully dealing with a ghost
  + The player can talk to the ghost. Depending on the ghost this may have no use or initiate some kind of dialogue where correct choices made by the player meet peaceful win conditions and incorrect choices damage the player’s health (for example)
  + The player can attempt to run. This should either be guaranteed escape or require a successful dice roll + combat skill vs ghost check, whichever seems better.

# Stage 2

* Put all the game’s items in their correct place, behind conditions if necessary
* Create a file that holds combat functions for each ghost, each of which handles the non-intro dialogue and all combat features for that ghost
* It should then (hopefully) be possible to deal with each ghost in the game as it will be intended in the final version, as well as find items in the correct places.
* Implement a basic tutorial
* Create a win condition for the game (all ghosts dealt with, most likely) that displays an ending based on how the player went about dealing with ghosts:
  + A good ending if all ghosts were dealt with peacefully
  + A neutral, standard ending if some were defeated through combat
  + A bad ending If the player used violence to beat all the ghosts

At this point the game basically works in full. If we then have time…

# Stage 3

* Implement the timer, and have the pizza reflect how cold it is when examined
* Full sound and music
* Etc.