- Keep a specific goal in mind
- Start from the beginning and work towards the end

---

1. Differentiate between playable characters and NPCs (why? so NPCs can have an AI module, drop-treasure module, boss, etc., but are ultimately the same)
2. Implement a system where one soul can attack another soul. ✓
3. Implement Soul death
4. Create a good damage/defense formula (for use with abilities!)
5. Implement items / equippables (using composition) ✓
6. Implement inventory system ✓
7. ~~Look into event handling for keyboard presses~~
8. ~~Rewrite linked list implementation for rooms as STL compatible~~
9. Implement rooms ✓
10. Implement item notifier/handler ✓
11. Implement information modules for rooms, items, and characters ✓
12. Add some kind of sensible item-hierarchy system ✓
13. Fix item system ✓
14. Flesh out the slots for equipping something

Details

1. Un-protect "equipment" in item.cpp
2. Prevent soul from targetting itself
3. Prevent notifier from notifying if registered listeners is empty
4. Prevent item from being equipped by more than one soul
5. Implement max-size for inventory
6. Ensure no new keys can be added in any of the maps
7. Ensure no two items can be picked up by the same Soul
8. Add a nullptr check for linked rooms. Respond with something like "You can't move in this direction!" ✓

9. Error-catching for items dropped that are not in the inventory
10. Items and souls **have** to be in a room. ✓
11. Ensure rooms, items, and souls are initialized so that seg fault doesn't occur when soul enters room that doesn't have info
12. Make everything const