Project 2: First Fantasy

Front-End Battle Logic and States

1. Start of Round/Character Select State:
   * Character sprites should have a visual indication that they are interactable. Some kind of “on Hover” event such as making it pulse, grow, flash, or create a boarder. At this point, they should not respond to input.
   * Characters who are dead (not alive/at zero hp) should also appear differently and be unresponsive until this state changes.
   * The Submit Button will also be grayed out and unresponsive.
   * The “Action” buttons should have the name of the action the corresponding character performs (if that character is a member of the party. The button should be grayed out and unresponsive otherwise.)
2. Select Target State:
   * When first selected, the “Action” button should change color and say “Targets?” In this state, the Characters should now be able to respond.
     + The Front End should request a list of valid targets for this character and action combination from the back end.
     + Preferably, all valid targets for the selected action should provide a visual indication to urge the user to click on them.
     + (In the future, the action button should be a drop down to select the action desired.)
   * Characters who are targetable should have a visual indication when clicked on to designate them as a target.
   * Characters who are not valid targets should not respond, or better respond in a way that suggests it rejects the user’s input.
   * This behavior continues until the maximum number of valid targets is reached.
     + (For the purposes of the MVP, it would be acceptable for one target, or selecting a target party would select the whole party. The targeting system does not define a option other than “one” or “all.”)
       - Be mindful that even targeting an entire party does not guarantee that each individual in that party is automatically a target. Casters may not be able to target themselves, or dead targets, even if they belong to a target party. Always use the returned Targets List to define clickable Targets.
     + The targets selected should behave as a queue. If the user selects more than the given number of targets, the first selected should be untargeted automatically. (Again, this would not be obvious or necessary in MVP, but including it in documentation to be thorough.)
   * Once the maximum number of targets is selected, the “Action” button will change color and say “Confirm!”
     + Once clicked, the “Action” button will now gray out and say “Ready!” This doesn’t mean it’s unresponsive.
       - Clicking on an “Action” Button in the “Ready!” state should begin the Target State again for this character (to allow the user to change their mind after submitting their decision.
       - This behavior should overwrite whatever else the user is doing (since people can rarely be counted on to finish what they start before double-backing on a previous choice.)
3. Once all available characters are in their “Ready!” state, the Submit Button will change color and become responsive.
   * Once clicked, the submit button will send the user choices to the back end, and will receive a string with the round results in return, along with any perinate state changes, such as marking a character as dead, or whether a victory was achieved.
   * The string will be appended to the battle log.
   * Then, all of the characters and states return to their initial settings to begin a new round of selections.
     + Or in the case of a victory or defeat, some indication of such and possibly redirected to another page.