

DESIGN PROPOSAL DOCUMENT

MECHANIC NAME:

MECHANIC TYPE: E.g. power-up, enemy, hazard

PLAYER NODE SELECTOR

BASIC MECHANIC

MECHANIC BEHAVIOR:

Treat this as if you are writing specs for an engineer to implement.

We will have 2 different node selection systems, depending on the input system (Console / PC). For now, just worry about PC, but make it easy for the programmer to switch the control scheme to console and vice versa

Make sure you choose the Faerie Circle / Node from `PlayerNodeManager.cs`

- `HashSet<BaseNode> m_nodesInRange;`
- That should contain all the possible candidates of Nodes you can choose

Obviously, do this in $O(n)$ time, since this will be called every frame

PC Input System:

1. Choose the Node closest to `MousePosition`
 - a. Be careful about world vs screen position, especially since this is 3D isometric perspective,
2. Have a detection radius around the `MousePos`
 - a. Ignore all Faerie rings outside the radius

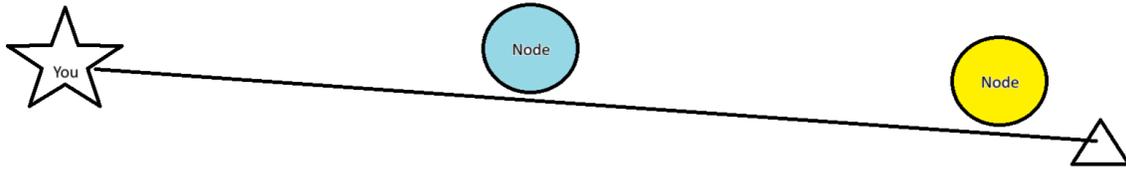
Make sure players feel like they don't have to click on a faerie circle, but still feel like they're going to the faerie circle they've chosen

Console Input System:

Look at `PlayerNodeSelector.cs`

1. Choose the Node closest to the direction of the joystick
 - a. Closest, not by distance, but by angle
2. Ignore any Node that is too close to the player
3. Pick the Node in this following example

DESIGN PROPOSAL DOCUMENT



- a. Notice that the player is aiming at the blue node, but because the yellow node is slightly closer in angle to the direction, they will choose the yellow node instead.
- b. Here, recognize this scenario, and choose the Blue Node