WindMill Node Project Documentation
Ron Cohen

Original plan

Designing Code Structure -
Writing down the essential requirement and features the assignment requires and translating it into unity features.

Research -
Researching similar systems and different features i’m not familiar with

Assets -
Building an asset library with all the required art to help with a visual representation of the system

Creating -
Building Prefabs for nodes and lines using the assets

Code -
Building code for each game object

Integration -
Connecting the different game objects using code

Testing -
Testing the system and tweaking it accordingly

First attempt

Creating Nodes with only unity based components
Each is made with a 2d circle and a script attached to each
I have given each node StageID(int) and State(string) (Locked, Open, Completed)
And gave each one a sprite with a public sprite within the script

Added a FindObjectOfType for a sprite renderer

Afterwards added a circle collider and a OnMouseClick method
And gave it debug.log to test it. Added a switch statement and used the state variable to switch between each state
The locked would not change the sprite and send a debug.log
The opened would change a sprite to completed
And the completed would not change the sprite and send a debug.log

Then i attempted to base the order of the nodes on the stage id alone
Quickly i learned it was not going to work

In start i made in if that made stage id 1 open according to the brief

Second attempt

Created a game manger that would start and spawn the nodes with a variable that would be used as a limit to a for method.
Game manager would give each object a name with a number to to determine their order

And i need to add a game object for the next node in each node

I wanted to try to keep everything in check so i created a public variable fo the current gameobject as well as the next one

I ran into a problem with the last node i tried to create because it didn't find any node after it. So i made it null for now using a bool that checks if there is a next node

Then when i tested it i realized i don’t have access the the next node spriterenderer
So i used getcomponent (tho i tried to avoid it)

I saw the the sprite changed but the state stayed the same so i created a method called set state the would attempt to change the state of the node but it didn't work

Third attempt

I tried to go back to the basic and base the system on linked list data type
Using nodes as the data type stored
Unfortunately i didn't have enough knowledge to use the references with a data type that not a prime data type and so i failed

Summery

It was a failed attempt but i learned a lot from it
Still got a long way to go and i will try to complete this project either way

1. Underestimate the complexity of the project
2. Spent too much time on getting the sprites to work rather than the code logic
3. Spend too much time on a bad attempt because i didn’t want to give up on it
4. While starting the project with a solid plan, everytime i wanted to create a new element i didn’t integrate it into the plan and made a mass