Brandon Coffey

# Project 03 Planning – Factorio Systems

For Project 03, I would like to recreate the Conveyor Belt and Automation System as seen in Factorio. The mechanic of my focus is the transportation of items from one place to another. This type of system must be dynamic and simple in execution, and I will be making it modular with scriptable objects.

The core concept of automating systems has always been engaging for me, as the challenge isn't solving the problem, but rather finding a solution within a web of other problems and solutions, and it's incredibly complex. I want to build a system that allows for complexity and interacting systems.

I believe that I can achieve this by making everything modular and only interact with what it needs to, as in its neighbors. Everything will be stored in a direct grid with its own stored coordinates, so it can use those coordinates to get neighboring systems. I have outlined in more detail the process I will follow to achieve a Factorio-like system.

As a side note, I will reference Satisfactory and Dyson Sphere Program as they offer a similar mechanic.

List of Core Elements that define the mechanics

1. A 2D grid-based building system where buildings and objects must match the grid
2. Conveyor belts that can move items in a certain direction
3. Buildings that accept items from the belts and output other items

List of Implementation Steps used to recreate the mechanic

1. Build a basic grid system where buildings can be placed down
   a. The grid will be a 2D array for now stored in a single object created at runtime.
   b. I plan to create a building template that can be placed down and have a rotation. This will be a script that all other buildings will inherit from.
2. Create a set of items
   a. I plan to use Scriptable Objects, which hold data easily. This is a simple process.
3. Create the conveyor belts
   a. For the belts, they will use the building template.
   b. The belts can store 8 of the scriptable object items.
   c. Using the defined rotation, they will try to detect a building in front of it, and then attempt to move items into that building.
4. Create the assembling machines
   a. The assembling machines will be a prefab consisting of a few building templates. Some will act as input and some as output and will be labeled.

    b. The prefab itself will contain the main script that holds a selection of items and uses a scriptable object to hold a recipe for items.
    c. The input section will reference the main script's scriptable object's recipe and only allow items of that type.
    d. The output will act as a basic conveyor belt that is given items by the script.

If time allows:

1. Create the inserters
    a. I plan to use the building template for the inserters.
    b. The script will detect the position behind and check for available items, and then detect the position in front and check for open spots for the detected items. If items available, it will move them.
    c. This is under "if time allows" because I am going for satisfactory's system of belts first. I would like to end with a dynamic system that allows either system.
2. Create player movement like Factorio's movement (2D, WASD)
3. Create a simple inventory system that allows the user to select buildings and place them down or items and drop them onto conveyor belts. The inventory will not keep track of quantity of items and will act as a creative menu.
4. Create a saving system that takes the grid-system on unload and saves it so when the scene is reloaded it places all the buildings and items back in their locations.