Create a document with a list of any mechanic, interactable, NPC, resource, etc. that you think you will require to finish the game.

Create a dependency list for what is required to complete this asset. Consider the following:

* What art is needed to make the asset
* What FX will it require?
* What sound?
* Animation
* What does it do? Describe it for the programming staff
* What variables do you need to edit on it? Speed, health, wait times, etc.
* How is it activated? Proximity, trigger, touch, etc.
* Does this asset interact with the GUI in any way? Interact button pops up on proximity? Update a resource bar?
* Does this asset affect anything in the game? Causes damage? Turns something else on? etc.

**Environment mechanics:**

**Moving Platforms:**

* **Art** – Platform model, texture (material dependent on stage).
* **FX** – No FX needed aside from sparks for dysfunctional platforms.
* **Sound** – Gear turning (passive background), “Ker-Chunk” on direction change.
* **Animation** – No animation needed, location transformations will be performed via script.
* **What it does** – provides a solid surface for both characters (depending on placement) to traverse route’s that they would otherwise be unable to alone/without assistance. Moving platforms help characters traverse large gaps or elevations with ease.
* **Variables** – Speed, point1Marker, point2Marker, etc.
* **Triggers** – Can be activated via button, switch, pressure plate, etc. or can be moved manually via console (usually housing a wheel or lever for the robot to use).
* **Other Interactions** – No other interactions are performed by this object. Does not cause damage, acts as if it were solid ground**.**

**Buttons:**

* **Art** – Button model, texture (material dependent on stage area).
* **FX** – Sparks for dysfunctional buttons, pulsating/solid light to show on/off states.
* **Sound** – Subtle beep or click upon press by a character.
* **Animation** – Button pressed down and back up animations needed.
* **What it does** – Provides a mean by which player characters can interact with environment mechanics, such as lights, platforms, doors, etc.
* **Variables** – Pressed Boolean, object it is attached to / triggers, button type (single press).
* **Triggers** – Can be activated via lever or other input module to power it. Acts as a checkpoint or follow-up action needed to complete objectives.
* **Other Interactions** – No other interactions are performed by this object. Does not cause damage, acts as if it were solid ground Can only interact with output modules (doors, lights, movable platforms, etc.).

**Lights:**

* **Art** – 3-4 fixture models, texture (material dependent on stage area).
* **FX** – Sparks for dysfunctional lamps.
* **Sound** – slight electrical hum for some fixtures, otherwise no sound is needed other than for switch on and off.
* **Animation** – No animation is needed.
* **What it does** – Provides a mean by which player characters can move through their respective parts of the environment; allowing them to go certain routes but barring them from others (light and shadow casting).
* **Variables** – on/off Boolean, width, cast height.
* **Triggers** – Can be activated via lever or other input module to power it. Non-togglable, so it can only be activated once
* **Other Interactions** – Provides routes for the robot to pass, while also creating shadows for the fox. Light damages the fox, so it uses the shadows created by it to traverse each level.

**Levers:**

* **Art** – Lever model, and a texture to accompany it.
* **FX** – sparks for dysfunctional levers.
* **Sound** – lever “Ker-chunk” on toggle.
* **Animation** – Lever switch on/off animation needed.
* **What it does** – toggles lights, doors, etc. on and off as a way of making puzzles more complex. Acts likewise to a button but is permanently togglable.
* **Triggers** – Is triggered by character interaction, permeant button bound not set.
* **Other Interactions** – Does not have any other interactions other than electrical output modules.

**Boxes**

* **Art** – A simple crate model, accompanied by a standard wooden texture with decals.
* **Sound** – wood on cement scratch sounds for when it is moving/being pushed.
* **FX** – Dust cloud on break and respawn
* **Animation** – No Animations needed for this asset.
* **What it does** – The Box is meant to give the player more options for completing puzzles and traversing the environment.
* **Variables** – Weight, Size, position
* **Triggers** – No Triggers
* **Other Interactions** – Interacts with weighted pressure plates in completing circuits. It can also interact with water and lighting; partially stopping their respective flow.

**Moving lights**

* **Art** – silhouetted Sprite / person model (Unity included). Lamp model
* **Sound** – Light footsteps.
* **FX** – The Lamp will be flickering.
* **Animation** – The Sprite will have a walking animation.
* **What it does** – The Moving light is meant to assist the player and hinder him. For the robot the moving light can help him get to place he can not get to normally, however for the fox the moving light will cause the player to take damage.
* **Variables** – Speed, Range, Size
* **Triggers** – The silhouetted will start pacing back and forth then the level is loaded.
* **Other Interactions** – No other interactions are performed by this object.

**Pulley Wheel Console:**

* **Art** – Crank wheel model with appropriate texturing required.
* **FX** – No visual FX needed.
* **Sound** – Iron squeaking (steel on steel) when wheel is turned.
* **Animation** – rotation of the wheel in both directions.
* **What it does** – Allows the robot to manually manipulate connected elements in the environment at will, while simultaneously controlling the fox, if the player wishes. This can range from a rotating light, to a drawbridge. Consoles with a crank wheel may also offer ways to solve complex puzzles.
* **Triggers** – When the robot interacts with the console, the wheel controls are activated/enabled.
* **Other Interactions** – No other interactions performed by this module.