Project Perfection

“You can’t kill perfection”

# 

[**Setting**](#_xldkztmg6vrn) **2**

[World](#_spgula3sarhu) 2

[Character](#_9clsu2gnek4d) 2

[Story](#_gc2whgmnsllk) 2

[Tutorial](#_uhsowml8pu8z) 3

[**Dungeons**](#_terykp9c1ujn) **3**

[Revisiting the same Dungeon](#_mzkjc3s72o7) 3

[Random Dungeon Levels (Technical)](#_i64oigniy0pq) 4

[Dungeon Enemies](#_i64oigniy0pq) 4

[Dungeon Bosses](#_grkoqs20t7lr) 5

[**Character Development**](#_p17eqw8ruqtc) **6**

[Level](#_va9b4vik1j49) 6

[Stats](#_fl36losht5n5) 6

[Inventory](#_frkebllogvnn) 6

[Abilities](#_7xknumwzgrxp) 7

[Movement](#_lngy0m3c06kv) 7

[Death](#_ojk68xhihss5) 7

[**Items**](#_5mgcguo4esg) **8**

[Helmets (Shields)](#_k2w8dazghj3f) 8

[Weapons](#_4vt1wpw7ucaq) 8

[**Loot tables**](#_o5xym9w6x4w1) **9**

[**Input**](#_orfrq7a4pyts) **9**

[Controller Support](#_czt2cnnlf7oy) 9

[**GUI**](#_rpz7273mrxyd) **9**

# 

# Setting

Genres: Sci-fi, Action Adventure

## World

The main game hub is a space ship, with the “P.E.R.F.E.C.T.I.O.N” machine, locker, cube, final boss door, and a door the player can take to go to nearby planets. This may later include a merchant or some way to gain some consumables, if those are added.

[Image: Map of hub]

There are 5 planets near the space ship, each with a different themed random dungeon. We can probably reskin enemies to some extent, so different recolours of an alien, different colours of a slime, etc.

1. Planet 1: A martian desert, overworld dungeon.
2. Planet 2: Water
3. Planet 3: Poison
4. Planet 4: Fire(tentative)
5. Planet 5: Jungle

Possible themes include: ice, water, fire, electricity, poison, technology…

[Image: Map of nearby space]

## Character

~~Either nameless, or possibly named a number.~~

Named “J” - the 10th creation/iteration in project P.E.R.F.E.C.T.I.O.N.

Built by the P.E.R.F.E.C.T.I.O.N machine, and tasked with gathering the parts of a mysterious cube.

[Image: character art concepts]

## Story

The player has been built as a part of project P.E.R.F.E.C.T.I.O.N, an attempt to build the perfect bioengineered person. They are tasked with gathering the parts of a mysterious cube for some mysterious AI. The player is not fully aware of what’s going on or why they’re gathering parts of this cube at the beginning. Once the cube is reassembled, the final boss can be fought. The final boss is the AI that originally tasked you with restoring the cube.

In order to make this more immersive we can add dialog from the partially completed cube, and AI, and possibly have some sort of a moral choice before the final boss where the “good” cube explains why you should not hand it over to the “evil” AI that tasked you with rebuilding it.

## Tutorial

(*This is up for discussion/design and needs concrete steps. I’m putting down some rough ideas I have here, based on our discussion - Alec*)

The tutorial will likely be in two parts: (1) aboard the ship [non-combat], (2) fighting the first gatekeeper [combat]

In part 1: introduce the player to the main game mechanics on the ship, possibly through the use of the mysterious AI.

* His mission
* Consequences of death, and how to mitigate them [locker/free stuff]
* How to go to planets
* How to equip items

In part 2: introduce the player to the combat system. At this point, this likely includes:

* Information on how to attack
* Information on health and shields
* Information on the gatekeeper mechanic.

# Dungeons

Each dungeon consists of a finite number of floors, determined before the dungeon is entered. The floor numbering corresponds to the depth in the dungeon, so floor #1 is at depth 1, floor #2 is at depth 2, and so on. As the depth increases, the difficulty should scale up.

Each floor has a starting/entrance room, and an exit room. The starting room contains a teleporter back to the hub ship. The exit room contains a teleporter back to the hub ship, or down to the next level in the dungeon.

The first floor of a dungeon has a gatekeeper, outside the front entrance to the dungeon. This gatekeeper is probably considered a miniboss, and might be a good way to introduce some of the mechanics to the dungeon in a controlled way.

The final floor of a dungeon has a boss. This boss guards/drops good loot, a map to the next planet, and a piece of the cube the AI is trying to rebuild. There should also be a teleporter out of the dungeon somewhere near the boss.

[Image(s): sample floors]

## Revisiting the same Dungeon

(*Adding this as something to think about - Alec)*

Is this desirable in some way? It would include possible restrictions such as not being able to generate another random dungeon between revisits, and a penalty to loot or money dropped.

(This is completely do-able, desirable I would argue not. As far as I know our core is combat and loot, not exploration - Dom)

## Random Dungeon Levels (Technical)

“Fluid room changes on a grid”

All level sections are build out of rooms.

All level sections are square and odd by odd sizes that can be expressed as integers.

The full dungeon is a n by m sized grid with a starting room spawned as an entrance. the entrance always has a teleporter out of the dungeon. Each room has 4 exit points.

By iterating over each exit point the dungeon generator can place additional rooms and fill in the areas that are used. If there is no space, then the door is removed and the adjacent room is not added. (Multiple attempts could be made to choose the correct room size)

Once the grid has been filled with rooms. A edge exit with no adjacent room will be chosen as the exit point. This exit point allows the player to continue to the next dungeon level or simply return to the base.

## Dungeon Enemies

Enemies can be spawned randomly from a spawner position that is part of the room. Each enemy has the following stats:

* Health(Techy enemies might have regenerating shields)
* Damage(Melee, ranged, instant, DOT)
* Defense(Basic damage resist. Bigger enemies have a higher value for this)
* Resistances(*This way an enemy is weak to a certain type of weapon and/or strong against other types -Spencer*)

We may want to tier enemies, so we have bigger versions of certain enemies that are stronger.

Attacking:

* Smaller enemies will be melee based or kamikaze
* Regular enemies will have ranged weapons of some kind(Whether that's guns or not)
* Large enemies will have heavy melee attacks or very hard hitting ranged weapons(Think LMGs and Gattling guns)
* Bosses are discussed below.

Probably not worth mentioning, but clearly stronger monsters drop stronger loot.

## Dungeon Bosses

For completionist sake: we mentioned the use of the same mechanic three different ways to make it the most satisfying/compelling. This idea applies to the boss design.

How Bosses Work: Bosses are typically stronger and smarter enemies with set mechanics that the player has to complete in order to defeat them. If they don't complete said mechanics, the player cannot win. Each boss with have three things:

1. One movement mechanic, unique to that boss and themes appropriately. Maybe can damage the player? Maybe not.
2. One shielding mechanic, something that hides the weak point from the player. Usually just something the player has to break.
3. One damage mechanic, that the player must avoid.

Bosses Concepts: Each boss will be themed to the dungeon it resides in. All the mechanics listed above will aid in that theming.

Boss 1| Mars-like:

Aesthetic:

1. Sandworm or Snake(Organic):
   1. Movement: Digs into the ground to disappear and pop back up away from the player.
   2. Attack: Giant beam that sweeps over an area.(Lightning?)
   3. Weak Point: After it finishes shooting the beam attack, it has to breath for a couple seconds. That’s your opportunity to fire at its one big eye.
2. Giant stone statue(Tech mixed with Stone):
   1. Movement: Disassembles and floats in a cloud of statue parts
   2. Attack: Attacks by assembling hands and arms and slamming the ground.
   3. Weak Point: When it slams, its head is in range for the player to shoot.

Boss 2| Water:

Aesthetic:

1. Octopus(I know cliche.Organic):
   1. Movement:
   2. Attack:
   3. Weak Point:

2.

* 1. Movement:
  2. Attack:
  3. Weak Point:

Final Boss Concept(*WORK IN PROGRESS*): The good AI cube and the player team up, fighting together to defeat the P.E.R.F.E.C.T.I.O.N. AI. The mechanics are similar to a regular boss, but with one caveat. The player has to protect the Cube AI while it disables the other AI’s shielding. The player isn't strong enough to release the shielding, so the cube helps him out.

(We can possibly take the four weapon types and make bosses out of them, plus the final boss being a mix of all of them - Dom Agreed.)

(What I put above is just a going concept. Not in love with it so it will be changed at some point. - Spencer)

# Character Development

## Level

This will likely be implicit, based on the power level of the equipped gear. That is, gear determines player power directly without any sort of experience system.

## Stats

The player, similar to monsters, will have a set of stats roughly as follows:

* Health (*Is this going to be a problem - without levels how does this increase? - Alec*). Intended as a “last resort” after shields are depleted. Hard to regenerate: likely only done through expensive/heavy items in the inventory, or uncommon/rare pickups after monsters die. Can be regenerated on the hub ship or through death. Starts at 100.
* Shield: An extension of the maximum health bar. Regenerates over time, when not being hit. Gained from the helmet slot, and varies by helmet worn.
* Defense: based on equipped helmet?. Reduces damage by a percentage.
* Damage: based on equipped weapon.
* Level/Power: A function of gear

[Image: concept art/mock up of stats page and maybe an item tooltip]

## Inventory

(*This still needs design work*)

We didn’t really discuss the specifics of this, other than that:

* The player should not be carrying around a lot of extra gear. At most, they should be carrying around about one helmet item, 4 guns, and limited amounts of consumables such as health packs.
* The player should be encouraged to quickly and easily switch gear. Thus, the stats have to be clear, and we want to highlight in a gold border items that are clearly better.

[Image: concept art/mockup of the inventory screen]

(Do we need an inventory screen? Can we not do it with just the UI -Dom)

I’m thinking something like Destiny where there is slots specific each type of time, like weapon slots, ability slots and item/ single use slots. - Spencer Gould

## Abilities

(*This needs design work)*

The player will likely only have access to a few of these at a time (through limited action bar slots). We can probably allow the player to change these between combat, or at the central hub if that’s necessary for balance reasons.

The only “ability” we have so far is a basic attack. We might not need a lot of other abilities if we also let the player bind consumables to their actionbar?

Abilities: Mostly based on a cooldown basis, using them has an illustrated cooldown effect that hampers you from spamming them. Can be movement based or damage based.

* Roll: Basic dodge move the character can employ.
* Teleport: Longer ranged teleport that takes a couple seconds to execute, and leave the player vulnerable.
* Assassinate: If the player gets behind the enemy and attacks without being detected, the player deals bonus damage.
* Overshield: Buff their shields to be very strong for a time period
* Bubble Shield: Place a stationary shield that can’t move but the player can pass in-between it. Takes damage either for a set time or until destroyed.

[We need some abilities, and concept art here]

## Movement

Movement will be done using game makers speed and direction variables that are built in. All movement will be dampened over time by friction.

Wall collisions will halt the player in the direction of the wall but allow them to continue moving in any free directions.

(Since we are doing the character side on, do we know what kind of collision box we want to use?)

## Death

Death causes the player to respawn on the hub ship, at the P.E.R.F.E.C.T.I.O.N. machine. The consequences for death involve the loss of items, and possibly money. Currently, this means the player drops their items, except for a helmet item and weapon stored in their locker.

# Items

Each item has a power stat, proportional to its strength. Higher power implies a strong item.

We might also want to give each item a colour rarity (gray, white, green, blue, purple, orange…) to indicate better items. Items stronger than the one equipped should probably show a gold outline on the tooltip. Items dropped are generated randomly on monster death (although stronger monsters may have guaranteed drops of a certain quality, and stronger monsters should drop higher quality loot in general).

## Helmets (Shields)

Style similar to borderlands, where shields have: Capacity (“max shield health”), Recharge Delay (seconds until recharge resumes, after taking direct damage), and Recharge Rate (shield health points regenerated per second). Different shields will have tradeoffs.

## Weapons

Weapons are either guns or physical weapons. They can be one handed or two handed. A player can hold 2 one handed weapons or a single two handed weapon. The weapons are then given one of the following ammo types:

* Rail
  + Rail weapons fire a single shot that does all of its damage to the single first enemy that it contacts.
* Plasma
  + Plasma weapons fire a slower moving shot that can pass through multiple enemies, doing damage to each of them once.
* Scatter
  + Scatter weapons spread their damage across a number of shots fired at random in a directional cone.
* Explosive
  + Explosive weapons create an area of effect at the first point of contact, doing damage to each enemy in the area.

All weapons have the following stats:

* Damage
  + Damage is the amount of damage per shot. Weapons show this as an integer before enemy armour is taken into account.
* Fire rate
  + Fire rate is the cooldown between shots in seconds (float)
* Capacity
  + Capacity is the amount of shots between reloads (int)
* Reload time
  + Reload time is the amount of time it takes to reset the capacity in seconds (float)
* Balance
  + Balance is a multiple that is applied to the damage of the weapon to help even out the difficulty curve.

Derived from the above:

**DPS:** We need this for comparing weapon quality

**Power:** The weapon’s relative strength, likely as a function of dps.

Weapons have infinite ammo, but need reloaded.

# Loot tables

(*This needs discussed, here are some ideas of how we might do this)*

One way we might do this is to give each monster a “power” attribute, based on how strong it is. To determine item quality, we might generate a random number which becomes the power stat of the item generated.

# Input

## Controller Support

Design with controller support right from the start (2 stick to enable aiming)

# GUI

Not discussed yet. Just adding a section header for it.