# 14号样本：砰！

## 运用膨胀的物理能力进行平台解密

## 目录

[14号样本：砰！ 1](#_Toc150715185)

[运用膨胀的物理能力进行平台解密 1](#_Toc150715186)

[1. 目录 2](#_Toc150715187)

[2. 修改日志 4](#_Toc150715188)

[3. 游戏总览 5](#_Toc150715189)

[3.1. 游戏概念 5](#_Toc150715190)

[3.2. 类别 5](#_Toc150715191)

[3.3. 特点 5](#_Toc150715192)

[3.4. 目标受众 5](#_Toc150715193)

[3.5. 游戏风格 5](#_Toc150715194)

[3.6. 终极目标 5](#_Toc150715195)

[4. 游戏玩法和机制 6](#_Toc150715196)

[4.1. 游戏玩法 6](#_Toc150715197)

[4.2. 游戏流程 6](#_Toc150715198)

[4.3. 谜题结构 6](#_Toc150715199)

[4.4. 目标 7](#_Toc150715200)

[4.5. 关卡流程图 7](#_Toc150715201)

[4.6. 机制 7](#_Toc150715202)

[4.7. 物理 14](#_Toc150715203)

[4.8. 游戏中的移动 14](#_Toc150715204)

[4.9. 战斗 14](#_Toc150715205)

[4.10. 游戏菜单流程图 15](#_Toc150715206)

[4.11. 游戏选项 16](#_Toc150715207)

[4.12. 重新开始和存档 16](#_Toc150715208)

[5. 故事、背景和角色——详见叙事设计文档 17](#_Toc150715209)

[5.1. 故事和叙述 17](#_Toc150715210)

[5.2. 世界背景 18](#_Toc150715211)

[5.3. 科技公司 18](#_Toc150715212)

[5.4. 角色 18](#_Toc150715213)

[6. 游戏关卡——详见关卡设计文档 20](#_Toc150715214)

[6.1. 关卡 20](#_Toc150715215)

[6.2. 教学关卡 20](#_Toc150715216)

[7. 交互 21](#_Toc150715217)

[7.1. HUD 21](#_Toc150715218)

[7.2. 控制系统 21](#_Toc150715219)

[7.3. 音乐、音效和配音 21](#_Toc150715220)

[7.4. 引导系统 22](#_Toc150715221)

[8. AI 23](#_Toc150715222)

[8.1. AI系统 23](#_Toc150715223)

[8.2. 敌对AI 24](#_Toc150715224)

[8.3. 非敌对AI 24](#_Toc150715225)

[9. 技术 25](#_Toc150715226)

[9.1. 目标硬件 25](#_Toc150715227)

[9.2. 开发硬件和软件 25](#_Toc150715228)

[9.3. 网络需求 25](#_Toc150715229)

[10. 游戏美术——像素画风 26](#_Toc150715230)

[10.1. 总览 26](#_Toc150715231)

[10.2. 角色 26](#_Toc150715232)

[10.3. 粒子效果 27](#_Toc150715233)

[10.4. 参考图 28](#_Toc150715234)

[11. 日程评估与项目规划 29](#_Toc150715235)

## 修改日志

|  |  |  |  |
| --- | --- | --- | --- |
| 时间 | 修改人 | 内容 | 备注 |
| 2023.7.11 | Han Haotian | 1. Create File | None |
| 2023.7.13 | Han Haotian | 1. 4.7. 部分机制 2. 4.8. 物理 3. 10. 游戏美术 | None |
| 2023.7.16 | Han Haotian | 1. 5.部分故事 | 1. 5.故事还缺少人物设计 |
| 2023.7.17 | Han Haotian | 1. 4.游戏玩法机制 2. 部分目录补全 | None |
| 2023.7.18 | Han Haotian | 1. 4.机制完成 2. 7.交互，8.AI系统 |  |

## 游戏总览

### 游戏概念

玩家将扮演

### 类别

* 2D
* 单人游戏
* 平台解密
* 线性故事

### 特点

* 物理
* 像素风格

### 目标受众

平台跳跃、解密类玩家。

### 游戏风格

像素、可爱+cyberpunk？

### 终极目标

A **short** game with **reasonable story** & **easy mechanics** but **complicated interactions** & high level of **completeness**.

一个**简短**的游戏，并且有着**合理的故事**、**简单但能创造复杂交互的机制**和**高完成度**。

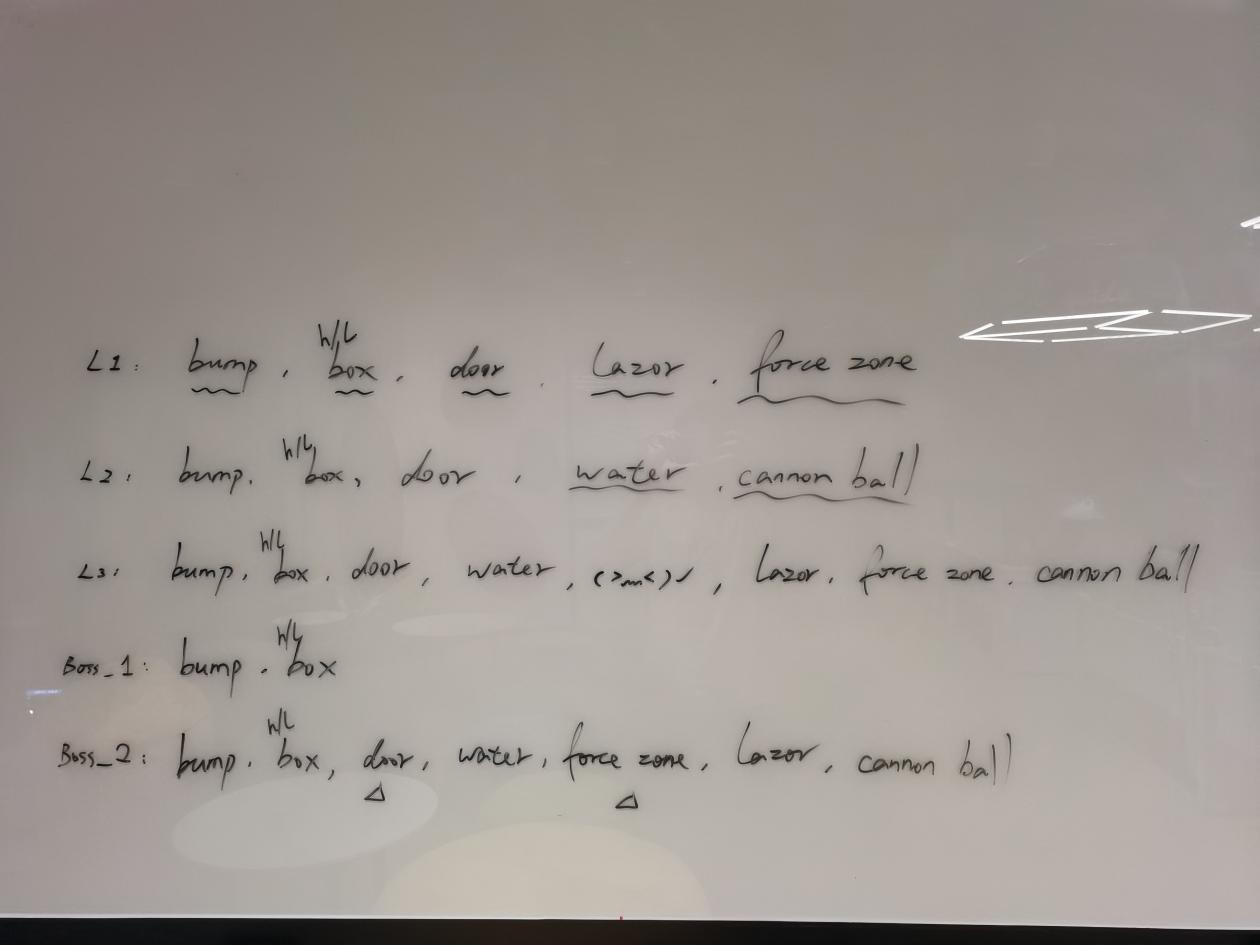
## 游戏玩法和机制

### 游戏玩法

玩家使用自身的膨胀能力与各种物体交互，触发门开关并进入。在触发过程中，玩家会遇到各种障碍，如敌人、镭射等，通过交互，以及各种物体之间的相互作用完成解密。

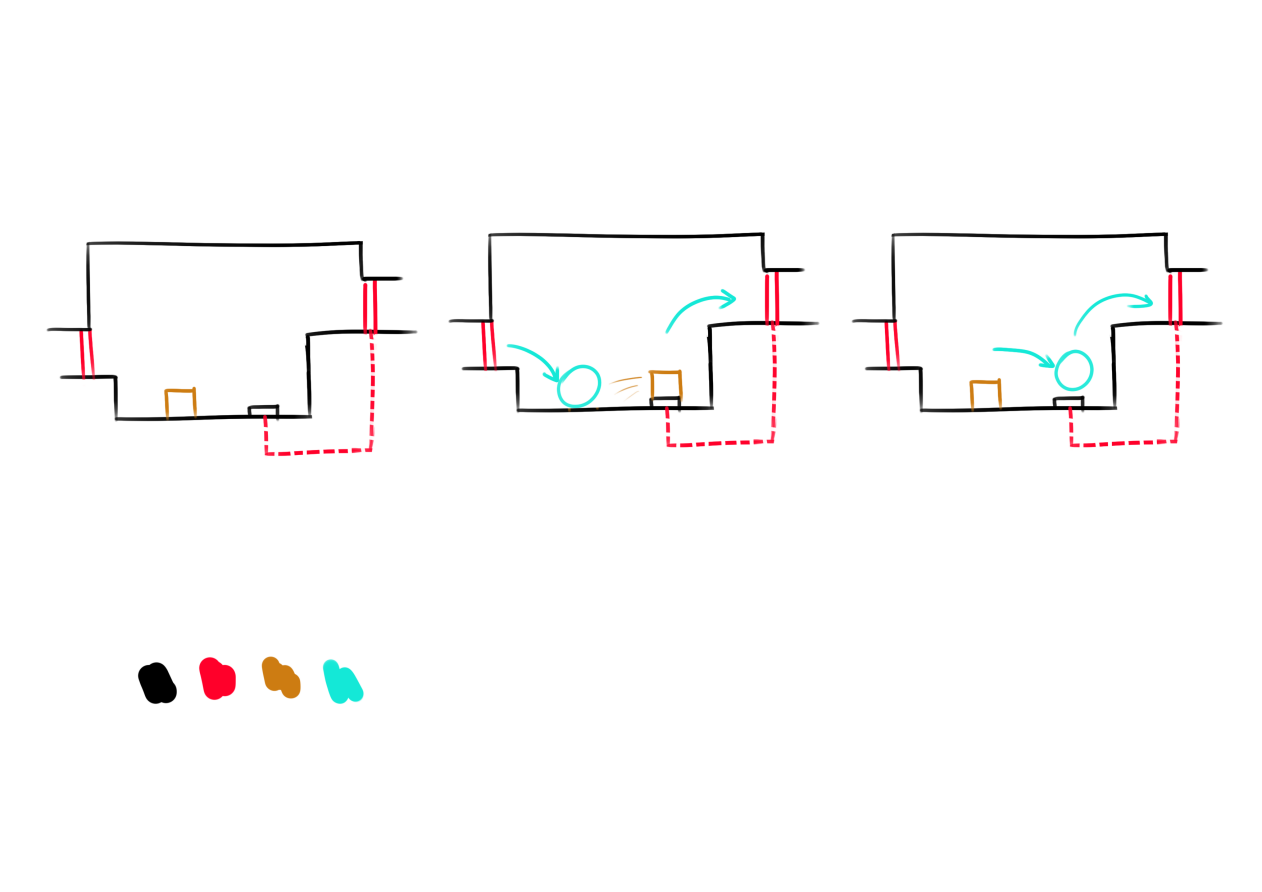
### 游戏流程

1. L1：第一幕，游戏的初期阶段，玩家将会被教学关卡的通过方式，以及一些最基础的机制的交互方式。
2. L2：第二幕，游戏的中期阶段，玩家将会接触到一些新的机制，并对原有机制进行复习。
3. LBoss\_0：第二幕，游戏中期阶段的高潮，但目的是打败玩家，玩家能使用的机制极为有限。
4. L3：第三幕，游戏的后期阶段，关卡是初期阶段的复用+变形，所有的机制都会涉及。
5. LBoss\_1：第三幕，Boss战的一阶段，目的仍是打败玩家，玩家能使用的机制极为有限。
6. LBoss\_2：第三幕，Boss战的二阶段，核心是游戏最基本的谜题结构，玩家将利用所有的机制打败Boss。



### 谜题结构

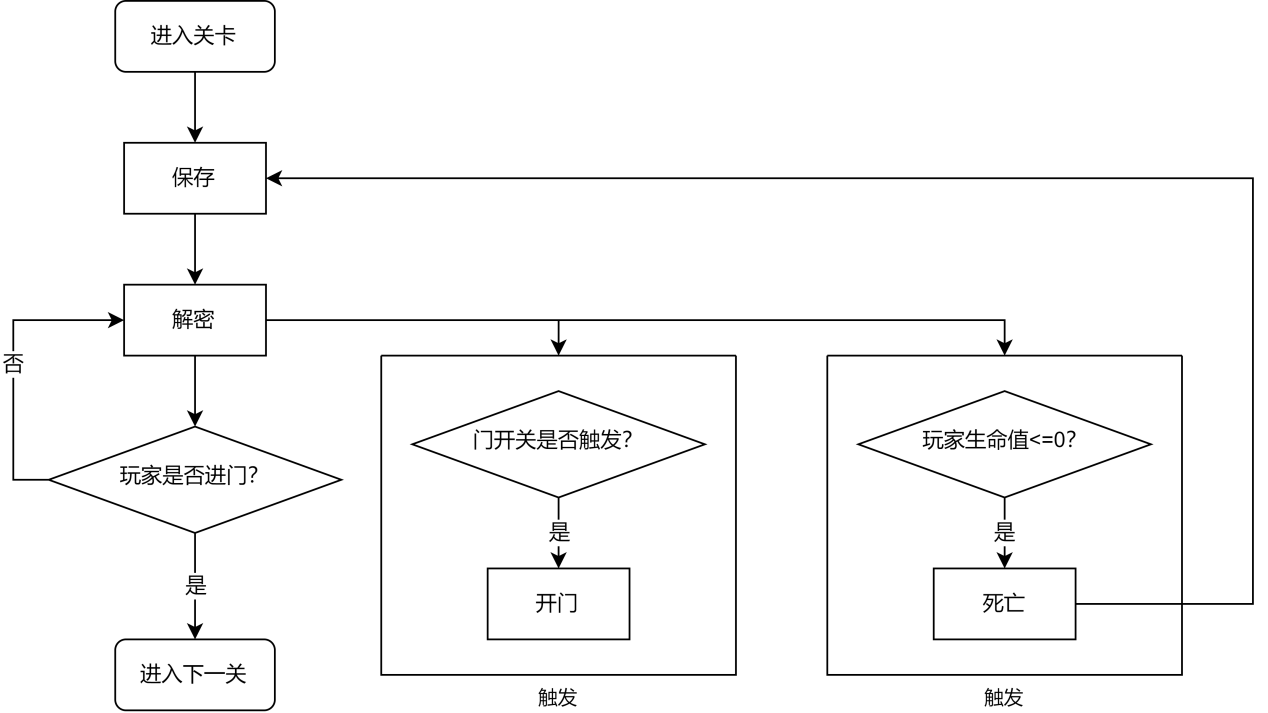
最基础的锁钥结构：每一关中都有门和重力感应开关（），玩家需要通过箱子等物体触发开关，并进入门（取决于不同情况，玩家自身也可以触发）。



### 目标

* 长期目标：完成游戏，逃离实验室。
* 中期目标：完成当前关卡，触发重力感应开关并进入门。
* 短期目标：使用膨胀能力与物体交互。

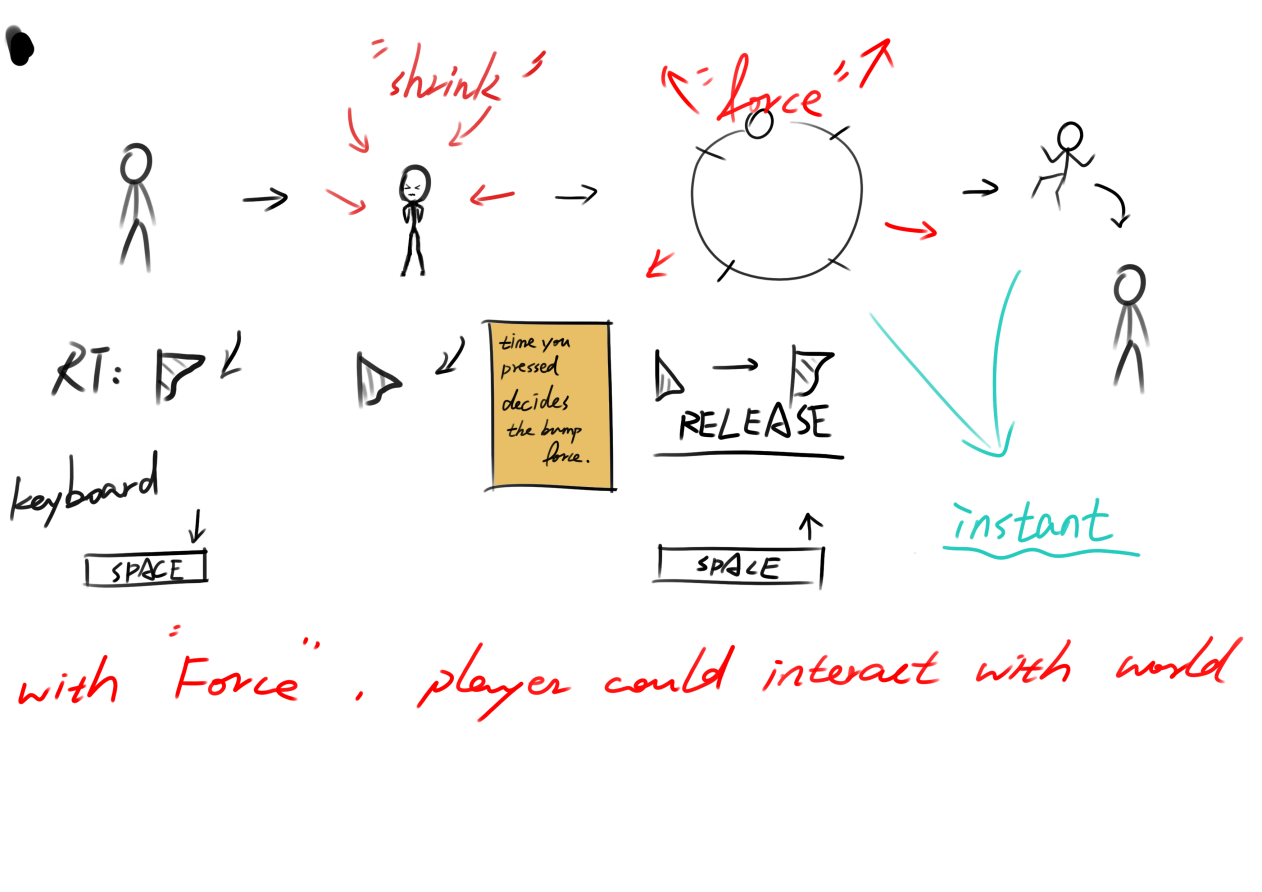
### 关卡流程图



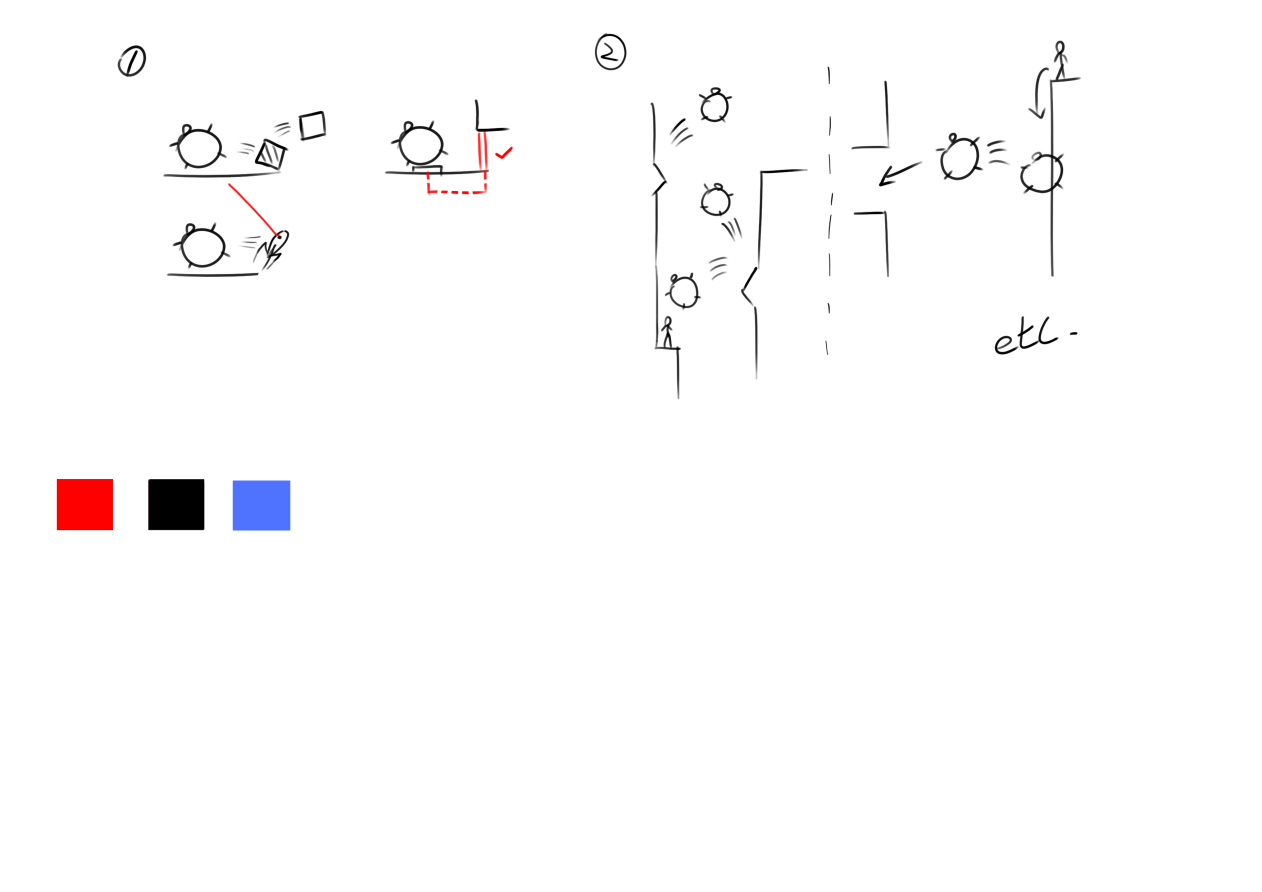
### 机制

#### 玩家机制

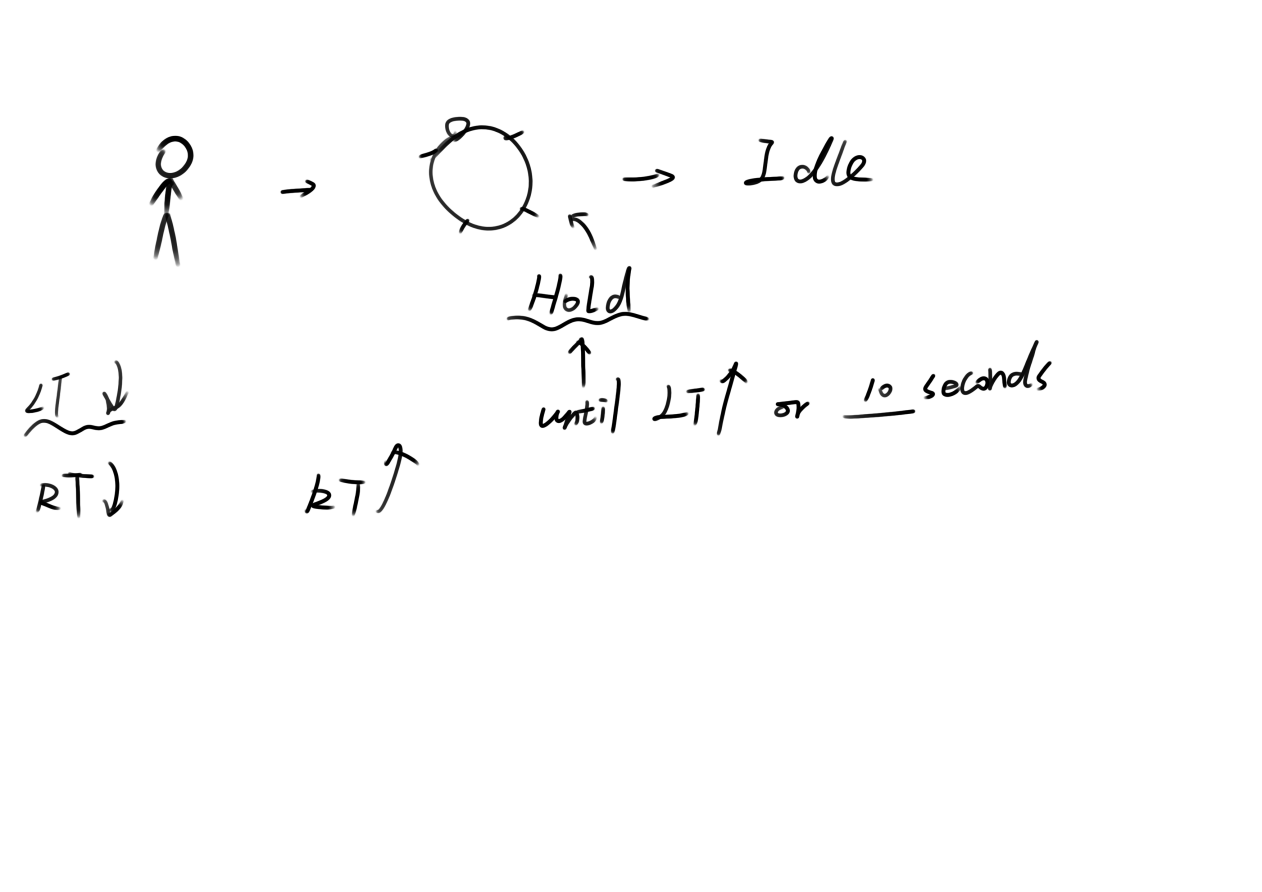
1. 核心机制：玩家膨胀自身，通过膨胀时产生的“力”与世界交互。



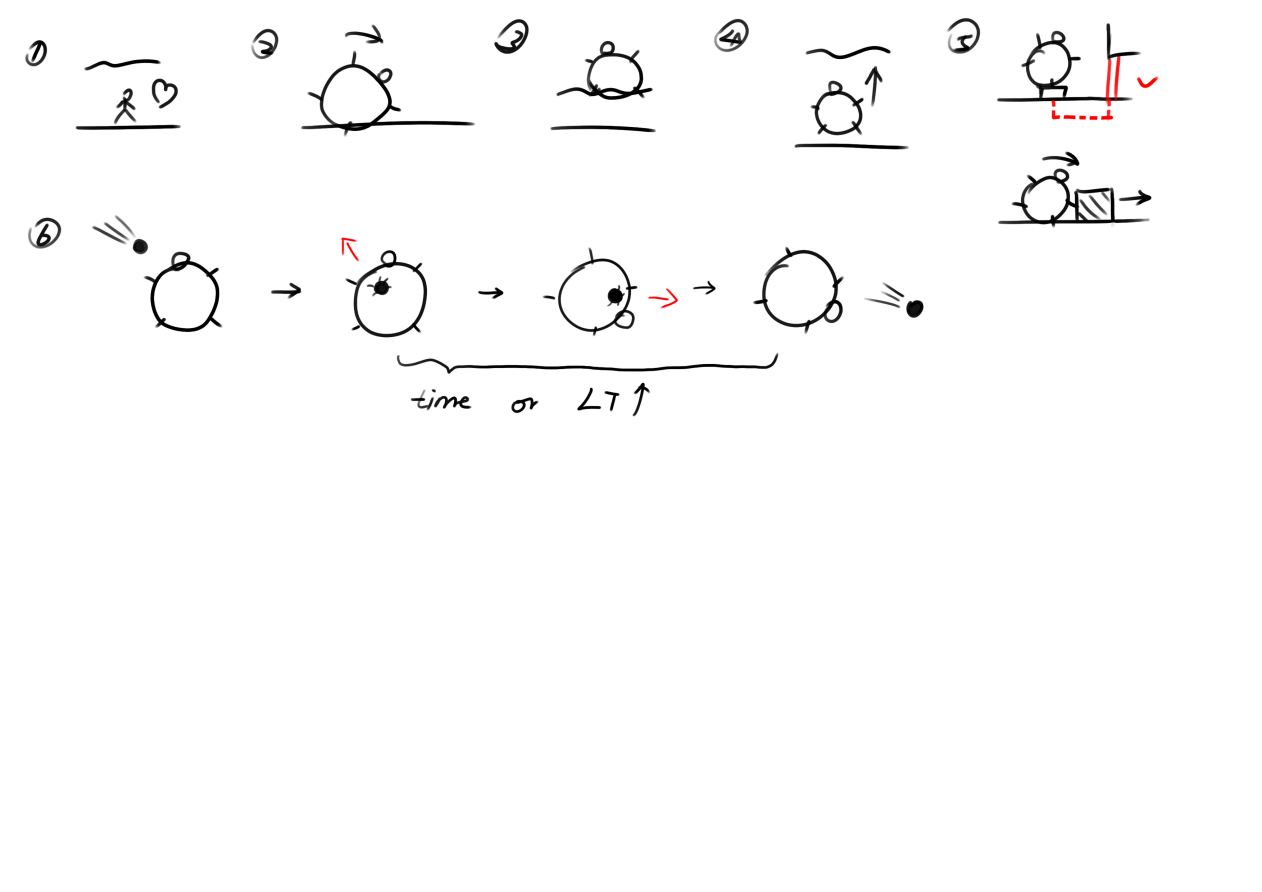
* 1. 触发：因控制方式不同，需要进行不同的操作设计。
     1. 手柄：按下RT，RT对应的float值对应主角缩小动画的程度，在此期间，玩家松开RT，将会回复到Idle状态。到达规定值后，主角缩到最小，开始计时，按下的时间对应膨胀后力的大小（有最小力与最大力）。松开RT，玩家膨胀。
     2. 键盘：按下Space，开始计时，同时动画以固定的较快速度自动播放到角色最小，按下的时间对应膨胀后力的大小（有最小力与最大力）。松开Space，玩家膨胀。
     3. 手机：（待定）。
  2. 交互对象：
     1. 与物体交互（如箱子、敌人等）；
     2. 与场景物体交互；
     3. 跳跃的替代。



1. 衍生机制——憋气：在憋气状态下，玩家在水下时，暂时不会受到伤害；膨胀时，玩家可以保持膨胀状态一段时间。



* 1. 触发：
     1. 手柄：按下LT进入憋气状态一段时间。在憋气状态下，按下RT，膨胀前期不变，膨胀到最大后，保持一段时间恢复或松开LT恢复。
     2. 键盘：按下H进入憋气状态一段时间。在憋气状态下，按下Space，膨胀前期不变，膨胀到最大后，保持一段时间恢复或松开H恢复。
     3. 手机：（待定）。
  2. 交互对象：
     1. 在水下时憋气——暂时不会收到伤害；
     2. 在地上时憋气膨胀——滚动移动方式；
     3. 在水上时憋气膨胀——以球形漂浮在水上；
     4. 在水下时憋气膨胀——快速上浮；
     5. 与敌人、物体等交互；
     6. 与投掷物交互。



#### 物体机制

##### 箱子（Box）

分为重箱子（Heavy Box）和轻箱子（Lite Box），两者属性和交互存在不同。

重箱子（Heavy Box）：

* 膨胀时受到的力较小；
* 可以在无其他力影响时，触发门开关；
* 不会被激光摧毁，可以挡住激光；
* 受力场影响；
* 在水中下沉；
* 不会被炮弹摧毁，炮弹击中它后爆炸。

轻箱子（Lite Box）：

* 膨胀时受到的力较大；
* 可以在无其他力影响时，触发门开关；
* 会被激光摧毁，无法挡住激光，摧毁后会重新生成；
* 受力场影响；
* 在水中上浮；
* 会被炮弹摧毁，炮弹击中它后爆炸，摧毁后会重新生成。

##### 门（Door）

* 固定物体；
* 门、重力感应开关均无法被摧毁；
* 重力感应开关检测到物体触发条件满足后，一段时间延迟后，门打开。

##### 镭射（Laser）

* 镭射可有固定物体或敌人发出；
* 固定物体镭射可以控制开关镭射，敌人

##### 力场（Force Zone）

* 固定物体；
* 力场由固定物体发出，可以控制开关；
* 对玩家、箱子、炮弹、敌人有特殊交互。

##### 水（Water）

* 固定物体，在场景中表现为一块特殊2D区域；
* 玩家对其有特殊交互；
* 对箱子（Heavy Box & Lite Box）、炮弹（Cannon Ball）、敌人（Enemy）有交互。

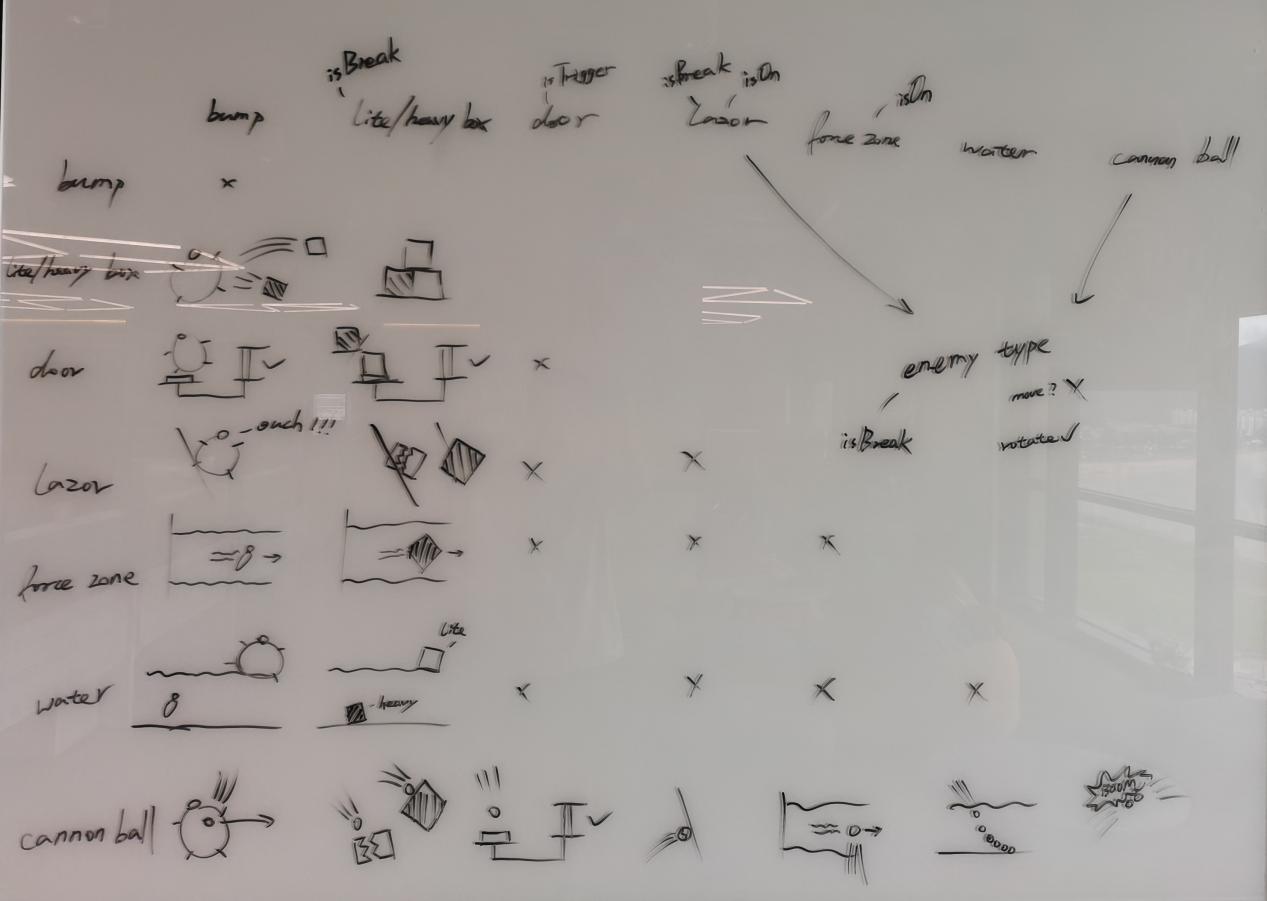
##### 炮弹（Cannon Ball）

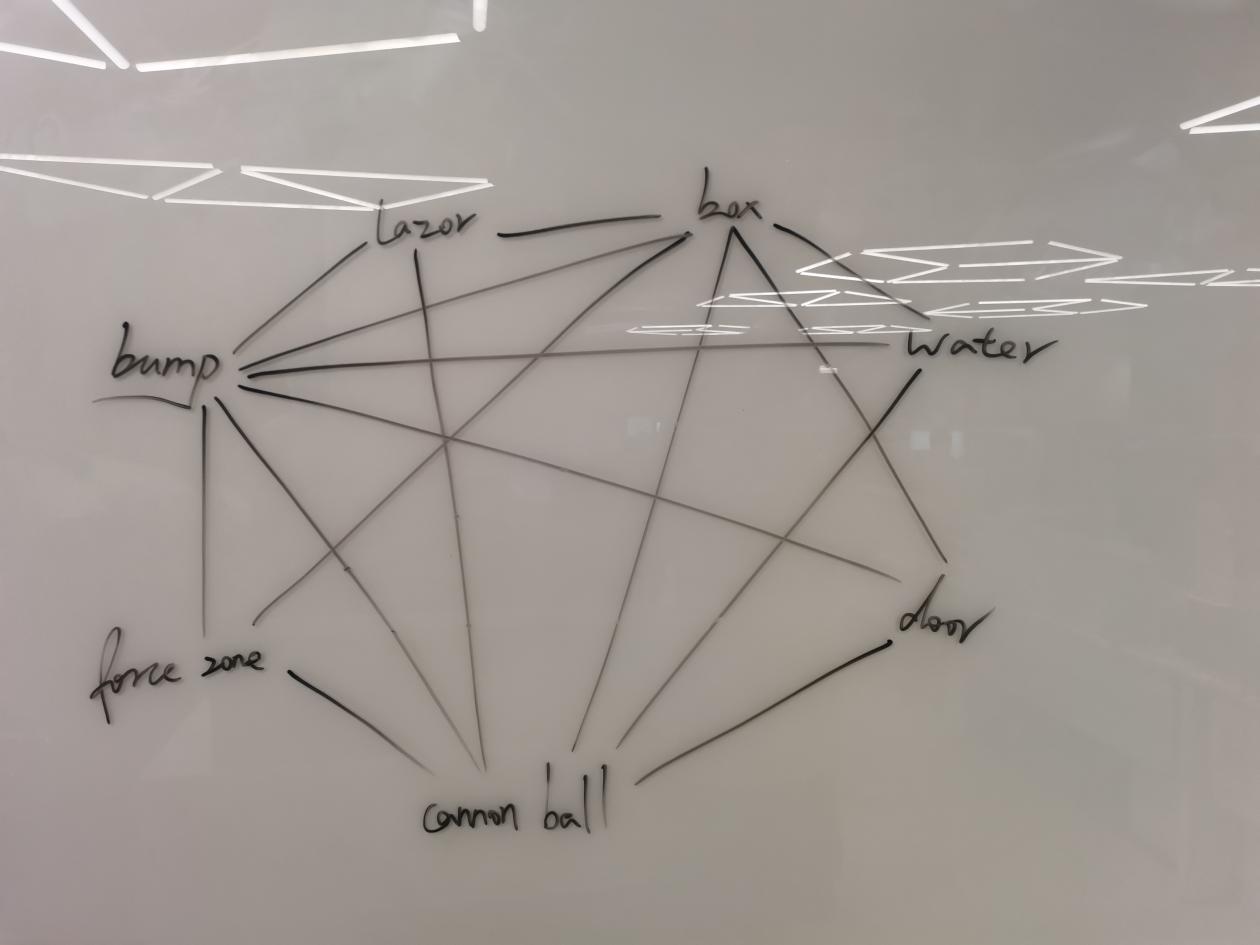
* 炮弹由敌人发出；
* 不受重力影响，但受力场、水影响；
* 玩家对其有特殊交互；
* 能摧毁轻箱子（Lite Box），触发门开关，不能摧毁重箱子（Heavy Box）；
* 碰到镭射（Laser）后，会爆炸；
* 两个炮弹互相碰撞后会爆炸；
* 能摧毁敌人。

##### 敌人（Enemy）

* 可以发射镭射或炮弹；
* 膨胀时受到的力较小；
* 可以触发门开关；
* 受重力、力场、水影响，在水中下沉，同时发射镭射或炮弹能力失效；
* 碰到镭射（Laser）、炮弹（Cannon Ball）时爆炸。

#### 机制相互作用表







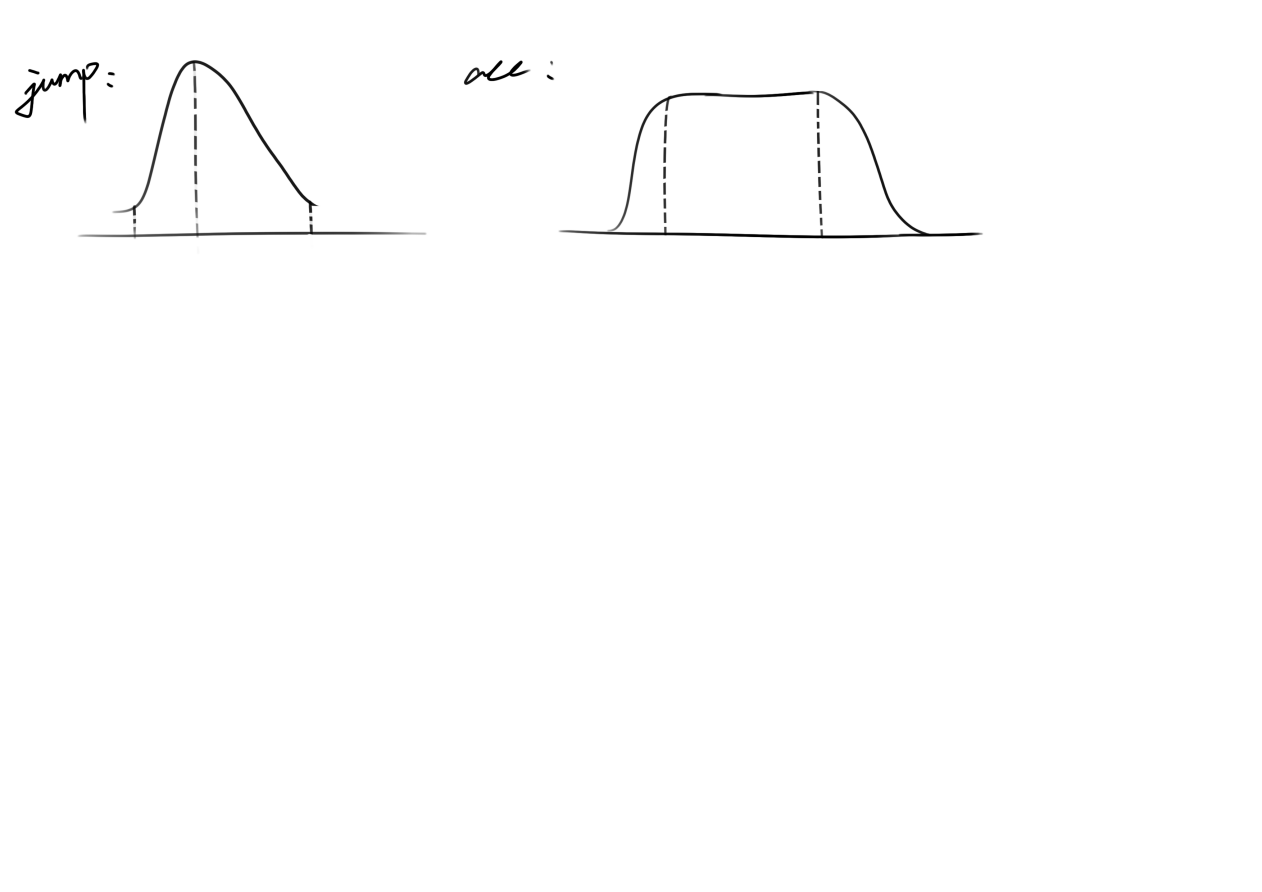
### 物理

* 游戏中玩家的物理均为代码模拟，不使用游戏引擎自带物理模拟。
* 通过射线检测替代引擎物理：射线检测与代码实现手感更好的2D平台操控器，参考素材代码：Corgi Engine。
* 玩家膨胀与障碍物相撞后的受力由玩家刚体AddForce实现，具体可见附件视频。
* 与可交互物体：并不使用物理引擎，使用碰撞体trigger进行检测，播放动画实现。

### 游戏中的移动

移动：加速时加速度高于减速时加速度，玩家更难停下（尤其表现在膨胀受力时）。

跳跃：减小玩家坠落时的重力参数，使玩家坠落的时间长于上升的时间，增强玩家膨胀的力量感。



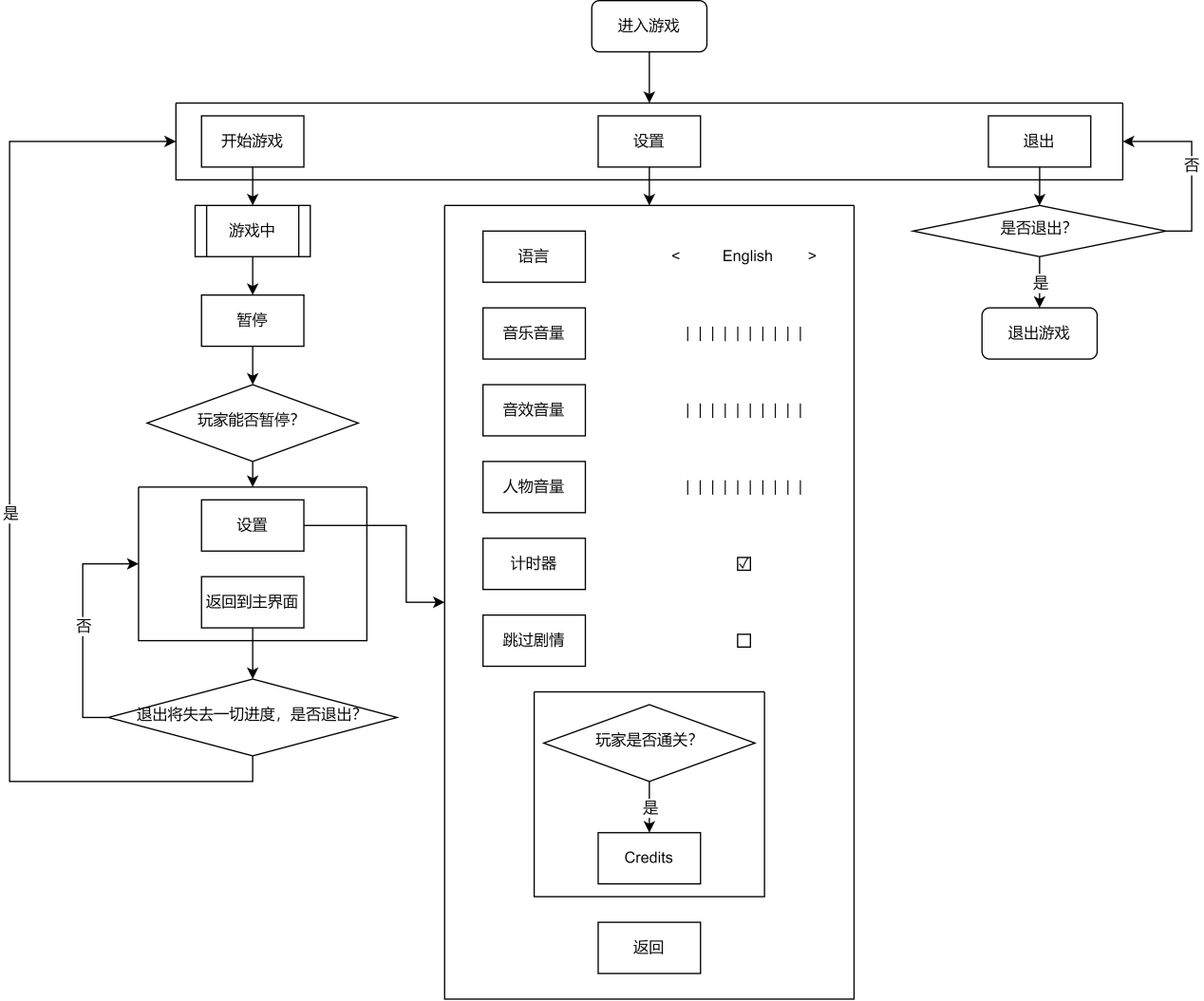
### 战斗

游戏敌人攻击方式主要有两种：镭射和炮弹。

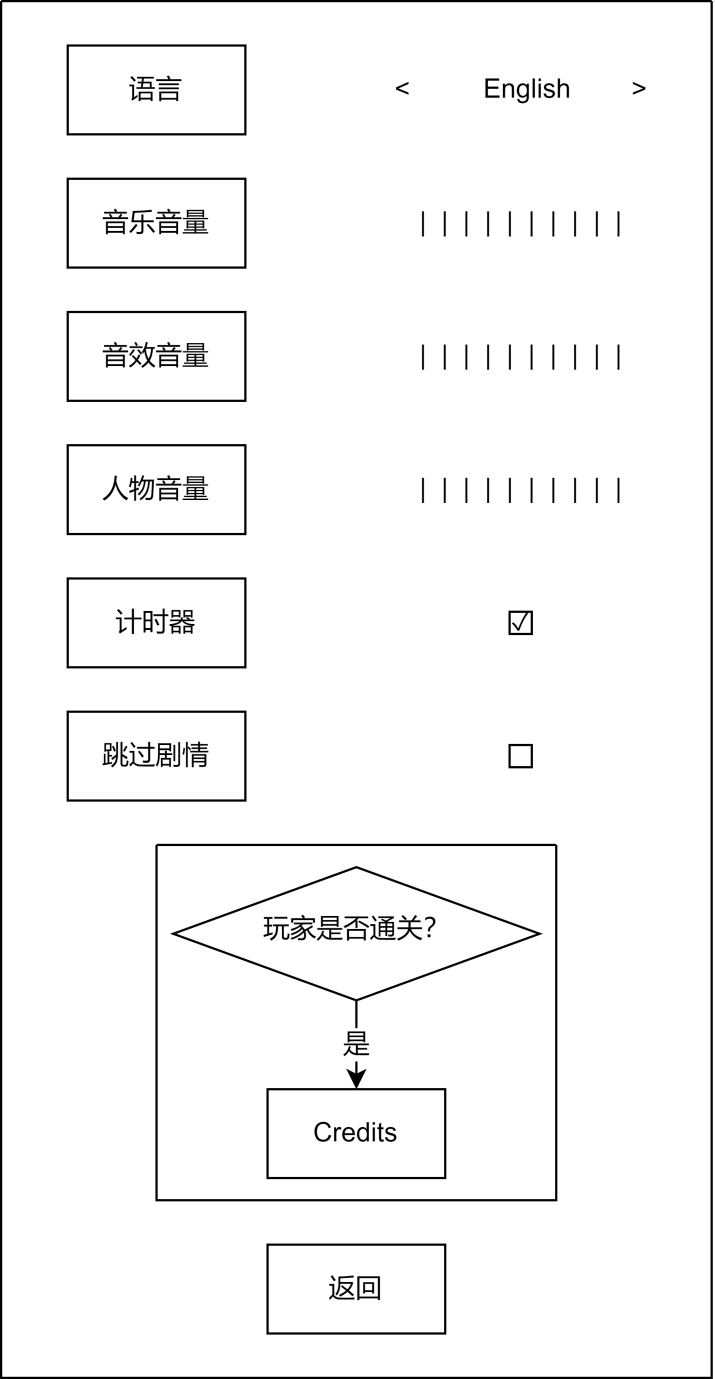
玩家的主要规避方式是依靠移动的躲避，但玩家也可以利用开关、反弹炮弹等方法消灭敌人。

1. 镭射：
   1. 存在形态：
2. 炮弹：

### 游戏菜单流程图



### 游戏选项



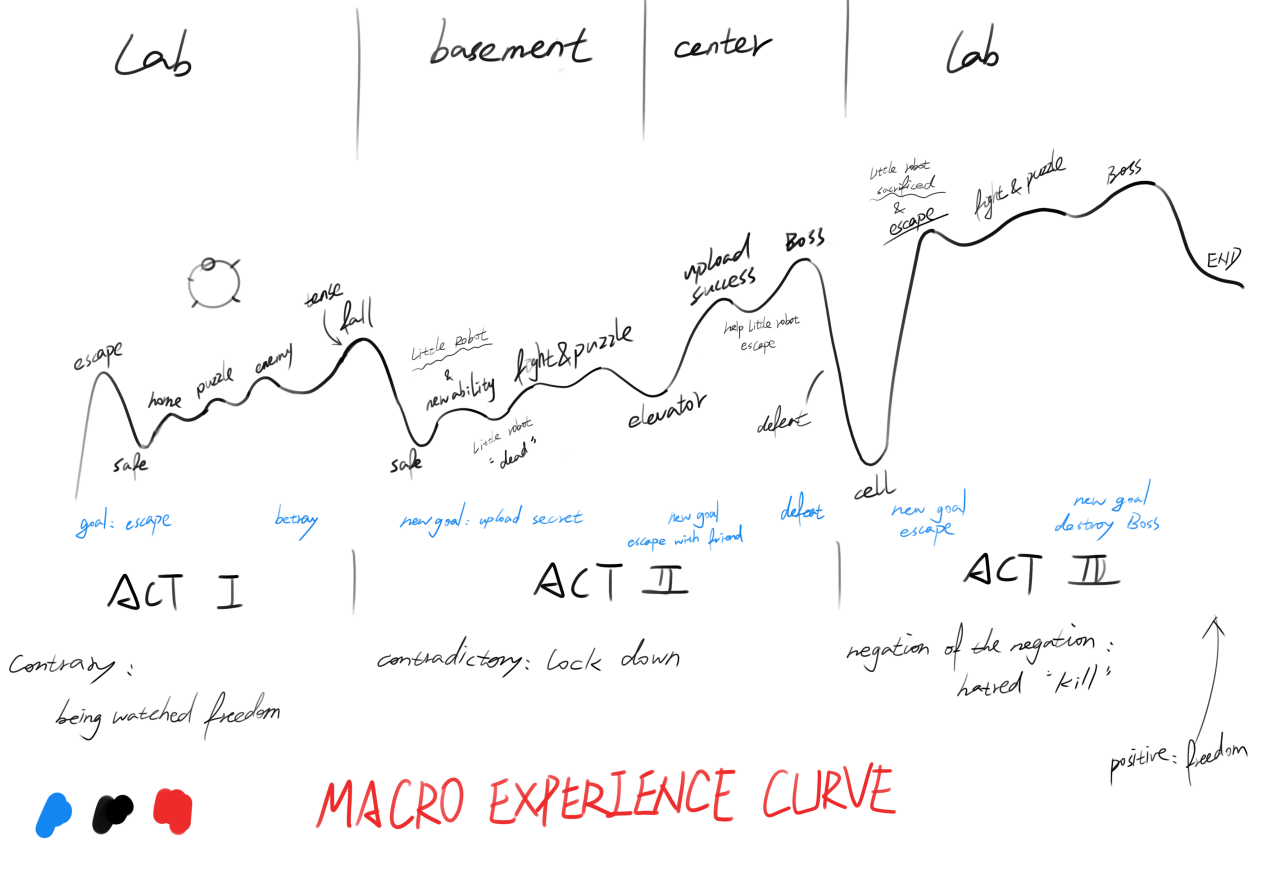
### 重新开始和存档

玩家在进入每一个房间后，都会触发自动存档。当玩家在当前关死亡后，会重置关卡并恢复玩家状态，放置玩家于关卡入口处。

* 游戏不设置退出后继续游戏的选项：游戏流程较短，需要体验连续。

## 故事、背景和角色——详见叙事设计文档

### 故事和叙述



1. ACT I：
   1. 高科技社会，科技基因公司私下绑架孩子进行基因改造。
   2. 主角（小女孩）从实验室醒来（记忆缺失），看到前面的人被推进一个屋子，传来被杀的声音。广播传来自大的语气：下一个就是主角，让主角听他，快跑。在逃离过程中，主角了解到自己是失踪儿童。
   3. 主角跟着广播，穿过几个不同的实验室（在实验室中遇到了机器人敌人），来到了大厅，大厅中站着几个和之前一样的机器人。广播说了一些不明意义的话，要求主角跟着那群机器人就好。
   4. 主角发现不对劲，跳入了一旁的管道。身后传来广播气急败坏的声音，要求机器人快抓主角，不然无法继续观察“样本表现”。
2. ACT II：
   1. 在下水道中，主角遇见了以前溜进来的“家用小机器人”。
   2. 主角获得膨胀的新能力，能飘过小机器人无法跨过的水道。小机器人将U盘扔给主角，希望利用主角上传秘密，活着逃离公司。同时杂物掉落，堵在了小机器人和主角中，身后追兵赶到，传来小机器人的惨叫声。
   3. 主角穿过布满敌人的下水道关卡，通过电梯来到总控室，上传了资料。
   4. 通过屏幕看到小机器人还活着，并且已经逃到了实验室门口，但却被大门挡住。主角通过广播和开关，帮助小机器人打开了大门。
   5. Boss（之前广播中的声音）来到总控室，打倒并囚禁了主角，小机器人也听到了广播中传来的声音。
3. ACT III：
   1. Boss发现了主角上传了秘密，将整个实验室搬到了船上，提取了主角的DNA，准备将主角杀死。
   2. 主角在监牢中醒来，无法逃脱。突然外面传来骚乱，小机器人拖着残破的身体帮助主角打开了牢门后牺牲死去。
   3. 主角又一次穿过了ACT I中的实验室，但这一次小机器人引起的骚乱使船体漏水，敌人也更加得多。
   4. 主角来到甲板上，见到了失去一切的Boss。在甲板上和Boss展开了Boss战。主角将要被击败时，甲板塌陷，来到有积水的下层，开展了第二场Boss战。
   5. 通过对能力和积水的运用，主角打败了Boss，坐着逃生艇离开了船体，在小艇上，主角拿出了从小机器人身上取走的芯片，芯片中传来了小机器人的声音。

### 世界背景

架空世界，2040年，社会科技尤其是生物科技得到巨大进步。社会与现在差别不大，但一家科技公司与政府暗中启动了人体基因改造项目，开始拐卖儿童并进行实验。

### 科技公司

科技公司，由科学家Friedrich创立，宗旨是：“拥抱科技，造福人类”。表面上是一家科技装置制造公司，私下里受政府支持，进行人体基因改造项目。因为担心告密，员工均是机器人。

公司沿海而建，有实验室、地下空间等，拥有船只，可随时将公司楼层搬至船上运离。

### 角色

#### 小女孩——主角

1. 背景：近期报道上走丢的小女孩，醒来后就已经躺在实验室的地板上了。
2. 描述：
3. 自觉欲望：逃离未知的实验室。

#### 小机器人——配角

1. 背景：
   1. 曾经为了保护主人（另一个小女孩），小机器人溜进了公司实验室，拿到了公司的秘密，只要上传到网络就能摧毁公司，但拯救主人失败，公司也发现并开始搜寻它。
   2. 小机器人躲在了下水道中苟延残喘，就这么经过了快一年，逐渐丧失了目标，只想活着离开这个地方。
2. 描述：在下水道奄奄一息但仍然嘴贫的机器人。
3. 自觉欲望：活着逃离公司。
4. 不自觉欲望：为崇高的事情牺牲自己，在心中释怀曾经的失败。
5. 人物弧光：
   1. 无助——困在下水道，失去希望。
   2. 利己——利用小女孩并逃脱至大门。
   3. 牺牲——在知道小女孩还关心着它并为它打开大门后，后悔利用主角，选择牺牲自己拯救这个和前任主人有些相像的小女孩。

#### 科学家Friedrich——反派

1. 背景：
   1. Friedrich，这个和尼采同名的人，怀着尼采“超人”的理念，决心要从生理上真正地创造超人。凭借其在生物科学上的成就，得到政府暗中的支持，建立了科技公司，开始了基因改造的研究。
   2. 他从小就刻苦努力钻研基因学，基本没有娱乐，将自己全部的时间投入到了“创造
   3. 超人”的事业中。但也深受着内心对成功无比渴望的煎熬。
   4. 渐渐地，Friedrich产生了极大的心理问题，他变得越来越恶趣味。而在得到政府的支持后，他认为自己离成功越来越近，逐渐在全是机器人的实验室中自大了起来。
2. 描述：
3. 自觉欲望：创造“超人”，得到成功与名誉。
4. 不自觉欲望：对“低等”人类的征服与蔑视。
5. 人物弧光：
   1. 自大，恶趣味——和其他样本一样，哄骗着主角，说要带领她逃离，但却只是为了测试。
   2. 怀恨在心——因为主角的一系列行为，真正严肃地看待主角，希望将她抓回囚禁。
   3. 癫狂，毁灭一切——在秘密被曝光，也失去了承载公司的船只后，真正地癫狂，愤怒了。

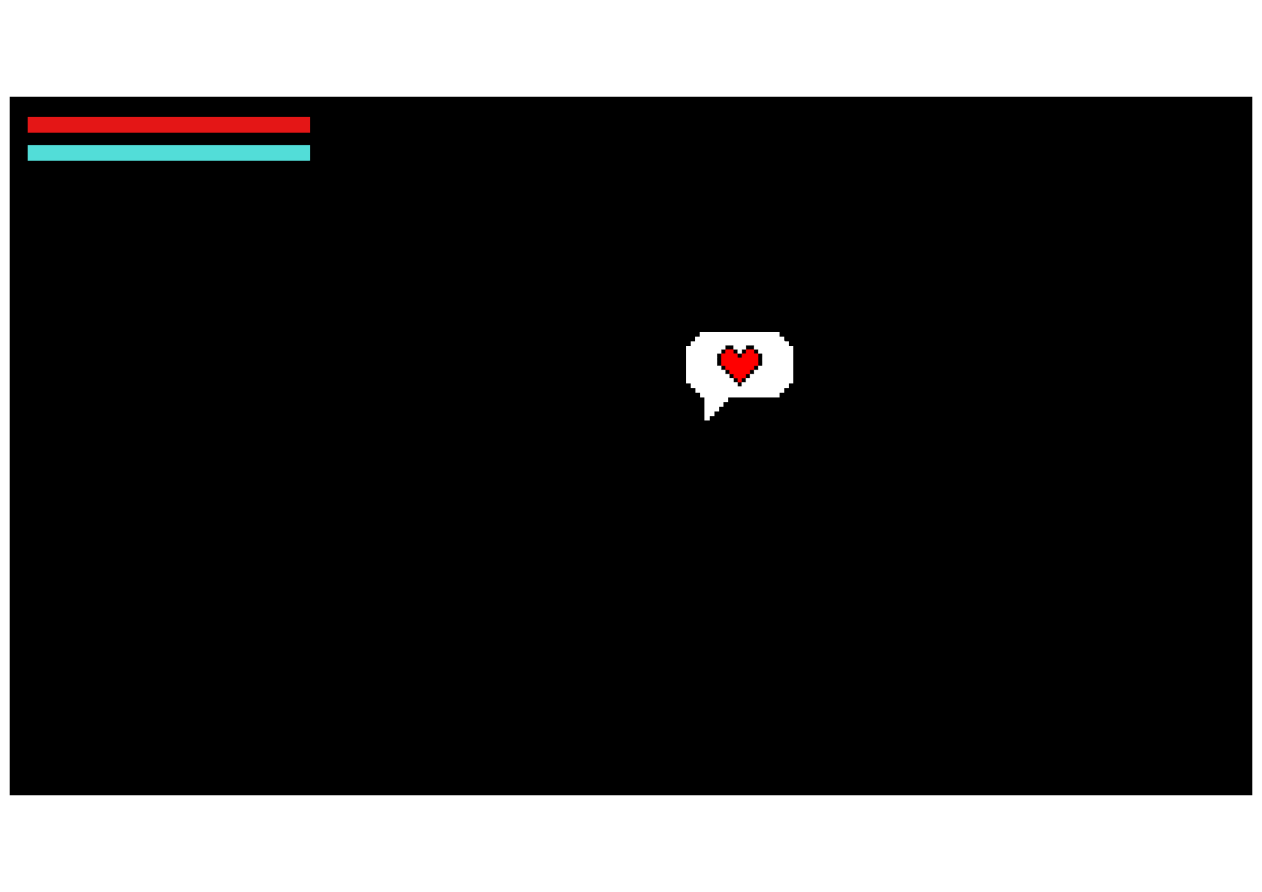
## 游戏关卡——详见关卡设计文档

### 关卡

### 教学关卡

## 交互

### HUD



### 控制系统

#### 手柄

#### 键盘鼠标

### 音乐、音效和配音

#### 音乐

#### 音效

#### 配音

### 引导系统

## AI

### AI系统

游戏中AI系统较为简单，复用Corgi Engine代码。

文档地址：<https://corgi-engine-docs.moremountains.com/ai.html>

### 敌对AI

#### Lazer\_Enemy

##### 属性

##### 决策树

#### CannonBall\_Enemy

##### 属性

##### 决策树

### 非敌对AI

#### 小机器人

##### 属性

##### 决策树

## 技术

### 目标硬件

### 开发硬件和软件

### 网络需求

单机游戏，无。

## 游戏美术——像素画风

### 总览

像素画风：

* 减少开发工作量——考虑使用AI+手改。
* 画面精度降低，玩家大脑自动想象缺少的动作，增加代入感。
* 简单的画风使玩家注意力放在机制和故事上。

### 角色

#### 小女孩

描述：

参考图：



#### 小机器人

描述：

参考图：

#### 科学家Friedrich

描述：

参考图：

#### 敌对AI

描述：

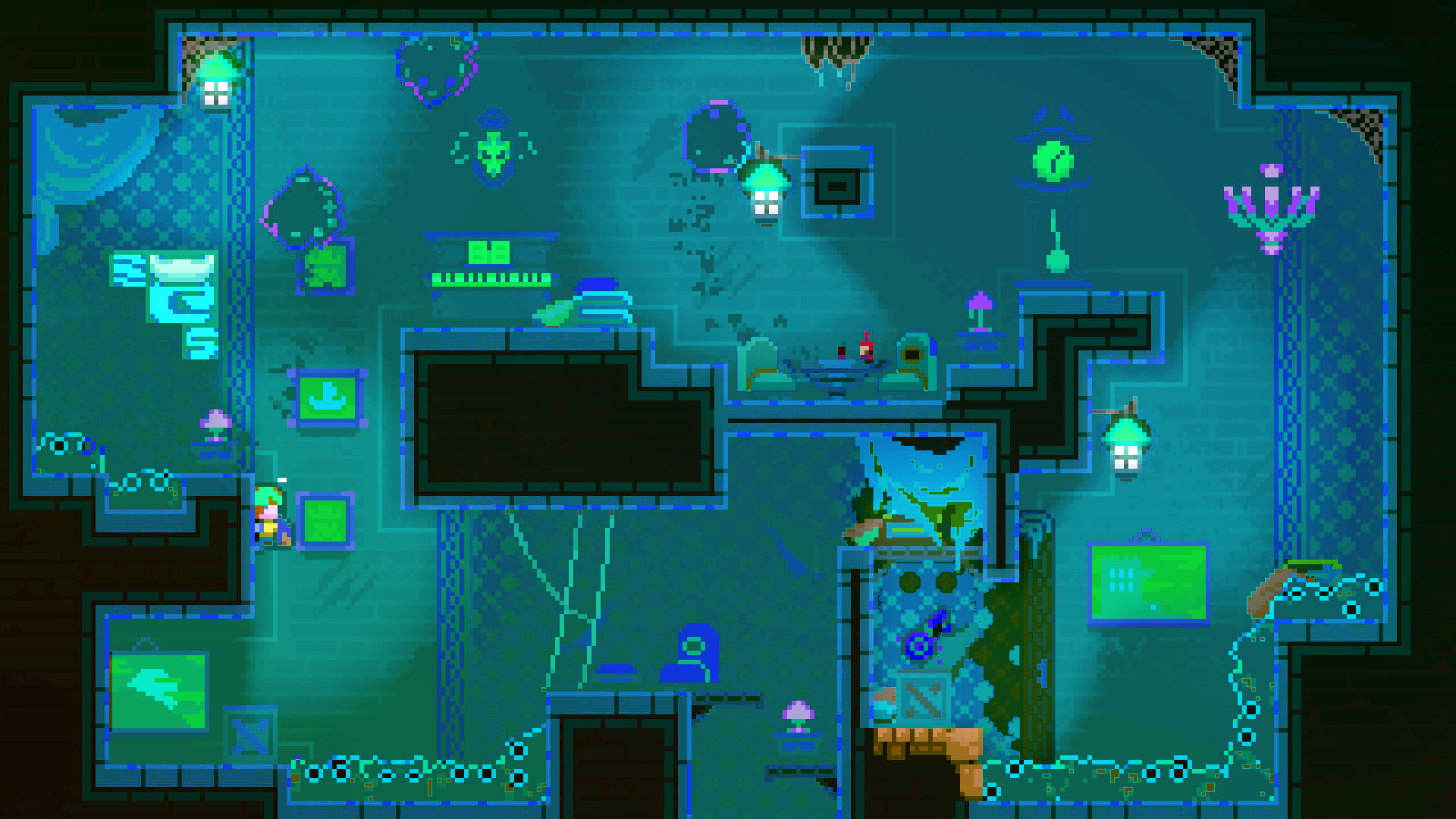
参考图：

### 粒子效果

1. 静态物体：如篝火、火把上的火焰、背景中的闪电等等玩家无法交互的物体。
   1. 使用Pixel FX Designer实现，本质上是图片组成的“帧动画”。
   2. Pixel FX Designer实现：减少工作量。
   3. Steam链接：<https://store.steampowered.com/bundle/16552/FX_Tools/>
2. 动态粒子：玩家受伤时喷出的血液、撞开门时的木屑等。
   1. 使用Unity自带粒子效果实现。
   2. 参考图：



### 参考图



## 日程评估与项目规划