# Hệ thống cấp độ

3 level và endless

Level : sử dụng bitmap

Chiều ngang màn chơi: 10 viên gạch, bên phải là điểm cùng mặt của ai đó

Chiều dọc: không quá nửa màn hình

Endless(ưu tiên thấp): level generator

Làm thêm level nếu thừa thời gian

Thắng : phá hết gạch

Thua: khi bị rơi bóng. Có thể cân nhắc thêm một yếu tố: vạch dừng

# Các loại Brick

Ngoại hình: hình chữ nhật

Brick: gạch, kim loại, vàng, kim cương. Tương ứng với lượng máu(health) là 1, 2, 3, 4.

Obsidian: loại brick không thể phá hủy

Explode: sát thương 1, máu 1, diện tích nổ giống hệt quân xe trong cờ vua

Stealthy Brick: máu 1, né nếu đạn đến gần, hồi chiêu 10s

# Các loại Power-ups

Để nhận power-ups, mình cần phải tiêu diệt những viên gạch. Viên gạch cấp 1 sẽ có tỉ lệ rơi ra ít nhất, cao nhất là kim cương.

Đạn xuyên gạch: 5s(ai code nên test để điều chỉnh).

Phân thân(Clone bullets): Vĩnh viễn tạo thêm 1 bullet

Tăng kích cỡ bàn nảy: 45s, tăng lên 150%

Tăng kích cỡ đạn: tăng sát thương tuyết, 10s

# Save and load

3 thứ cần lưu:

Array 2D

Trạng thái: vị trí, hướng bay viên đạn

Buff: loại buff và thời gian

Vị trí của thảm

Bảng xếp hạng

# Multiplayer

Làm sau

# Thiết kế code

Sử dụng như trong yêu cầu

### GameObject và các class sơ khai:

Bình, làm trước trong tối nay. Mọi người hạn là hết thứ 4 tuần sau

### Paddle, Ball: Diễn

Paddle: nhận input người dùng để di chuyển. Paddle là vật bị động, sẽ không kiểm tra va chạm

Đã có file Ball.java trong dự án. Cần kiểm tra va chạm với 4 cạnh màn hình và Paddle, tạm thời chưa cần kiểm tra với Brick

### Brick: Cường

Thiết kế hệ thống bitmap, Brick.java. Brick là vật bị động, sẽ không kiểm tra va chạm

### Power-ups:

tô đức anh, để sau

### Lưu ý khi code

Sử dụng github desktop để download repository của chúng ta về máy mình

Quy định commit: không giới hạn số lần commit, nhưng mỗi lần commit đều phải ghi title, và nội dung(không bắt buộc). Không có quy tắc nào cả, viết gì có ý nghĩa là được

Không được tự ý CHỈNH SỬA code của đồng nghiệp, nếu cần, hãy báo trước

Ngoại lệ đối với hàm main() của GameManager là nơi mọi người có thể CHỈNH SỬA thoải mái, test, logic, tạo object gì đó…. Không động vào gameloop(), GameManager’s contructor, GamePanel…. chỉ duy nhất hàm main()

Kiến thức áp dụng: java cơ bản và xử lý 2D, kế thừa. Để biết một class có nên sử dụng kế thừa hay không, hãy trả lời câu hỏi “class này trong thực tế có sở hữu toàn bộ thuộc tính và hành vi của cha không?”

Khi bắt đầu code, nên khai báo tên các hàm trước, nội dung(logic bên trong) để trống

Nếu muốn test chương trình, cách duy nhất được phép là chạy file **build and run.bat**, chương trình sẽ tự động biên dịch(tất cả file .class sẽ vào folder class). Không dùng button RUN trong IDE, không tự ý chạy lệnh terminal