

# PROJECT

## Proposal

Road Rash is the game that was a part of my childhood. The game based around a reckless motorcyclist who races against other around the city and can do anything, similar to open-world gameplay. The character is chased around by police if ever commits a “crime”. Meaning by running over citizens or kick, punch, push other racers. The character then can choose to fight back the police by mentioned actions.

However, the game never focuses on the citizen AI since they will just keep on getting run over by the user without fighting back. I want to implement a citizen AI where it can choose to throw rock to the user upon running by, by predicting user’s location at X time.

## Issue

The challenging part is to design a “racetrack” for the user to race on. The user is then needed to follow that racetrack to reach the destination point. This project will be a combination of Agent Marksmanship task, Soldier on Patrol task and some Agent steering task with a little bit of advanced character designing using python. However, player needs to be able to control user’s character with some simple “go forward”, “go backwards” functions.

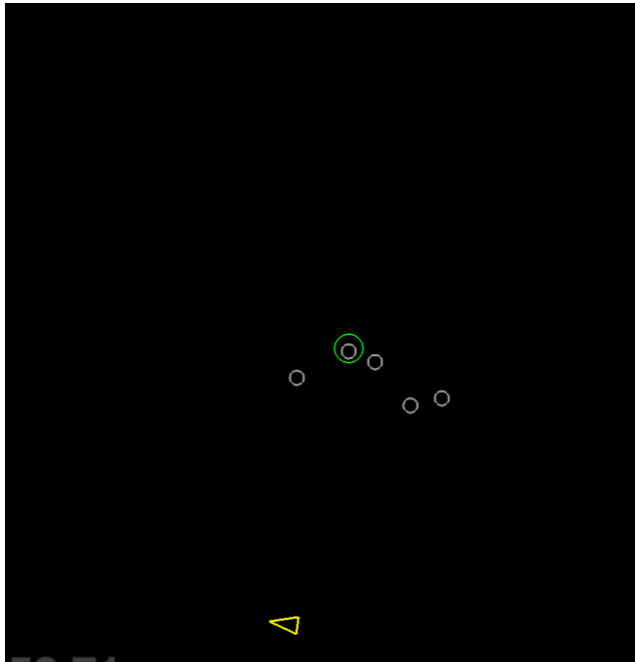
## Plan

Simplistically, there will be a road created for the user to race on. The citizen will wait at a location. When distance from user to citizen is X, the citizen will throw rocks at the user at a fixed velocity. User will need to choose, based on self-speed and the rock’s speed, to slow down or speed up to avoid the rocks. If not, user will lose a small amount of health.

```
def fire(self, target_pos):
    enemy_pos = target_pos
    if self.world.hunter.aim is True:
        bullet_speed = 1000 if self.mode in ['Rock'] else 10
        target_pos = self.aim()
    if self.mode == 'Rock':
        for i in range(1):
            self.world.add(RockBullet(self.init_pos, enemy_pos))

class RockBullet(Bullet):
```

```
def __init__(self, firing_pos, target_pos):  
    Bullet.__init__(self, firing_pos, target_pos +  
        Vector2D(randrange(-100, 100), randrange(-100, 100)))  
  
    self.radius = 5  
  
    self.velocity = 1000  
  
    self.velocity = randrange(18, 22)
```



## Result

It is expected that the user can control itself to avoid the rocks. However if the user is hit, it will change colour to indicate the damage taken.