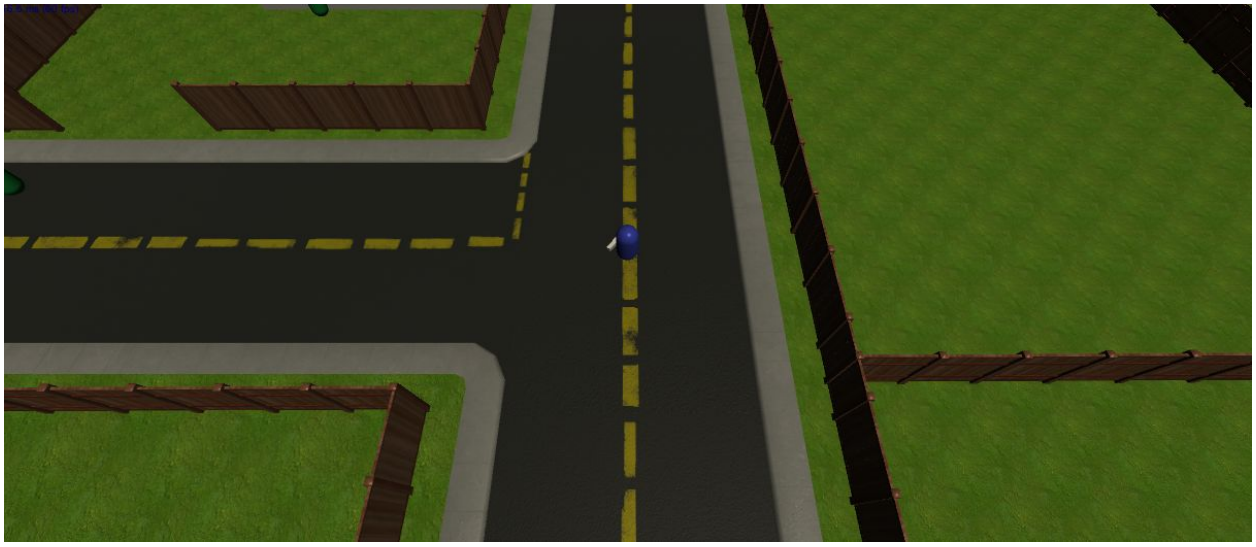
Technical Document - Strain

By Iain Dowling and Andrew Giannopoulos

Player view & Camera

Camera sits above the player at a slight angle, looking directly towards the player.



As the player moves the camera will follow, so that it remains at the same position relative to the player. As the player enters buildings or moves behind objects, the buildings/objects will become transparent so that the camera does not have to move to provide a better view.

This type of camera will allow the player full 360 degree vision of the area around their player character. This way the player can see, aim and shoot at all enemies approaching, no matter what direction they are approaching from.

Class list

NPC

Class that contains both the code for the zombie and human behaviour. Also manages interactions between NPC's, such as when infection occurs and when/how humans transform into zombies.

Chromosome

Container class for the information that makes up the virus strains.

ZombieSpawner

Manager that subtly controls populations based off current human and zombie populations, using concepts from the lotka volterra equations. Actually will only spawn initial zombies, then will spawn humans in certain areas as the game progresses to maintain appropriate populations.

GunClass

Holds the basic information for most projectile weapons, allowing its generalized use in prefabs.

LootSpawner

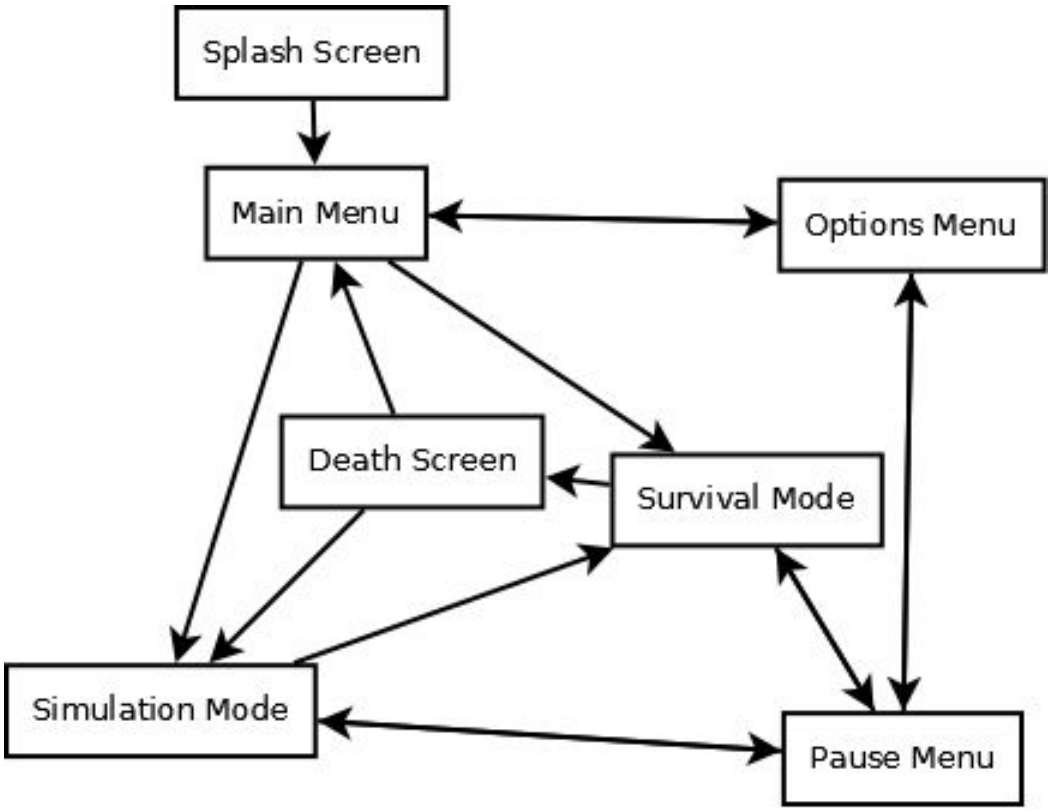
Invisible objects placed on the map that will semi-randomly spawn loot, but will be influenced by factors such as zombie strength in the area and human/zombie population ratios.

PlayerInterface

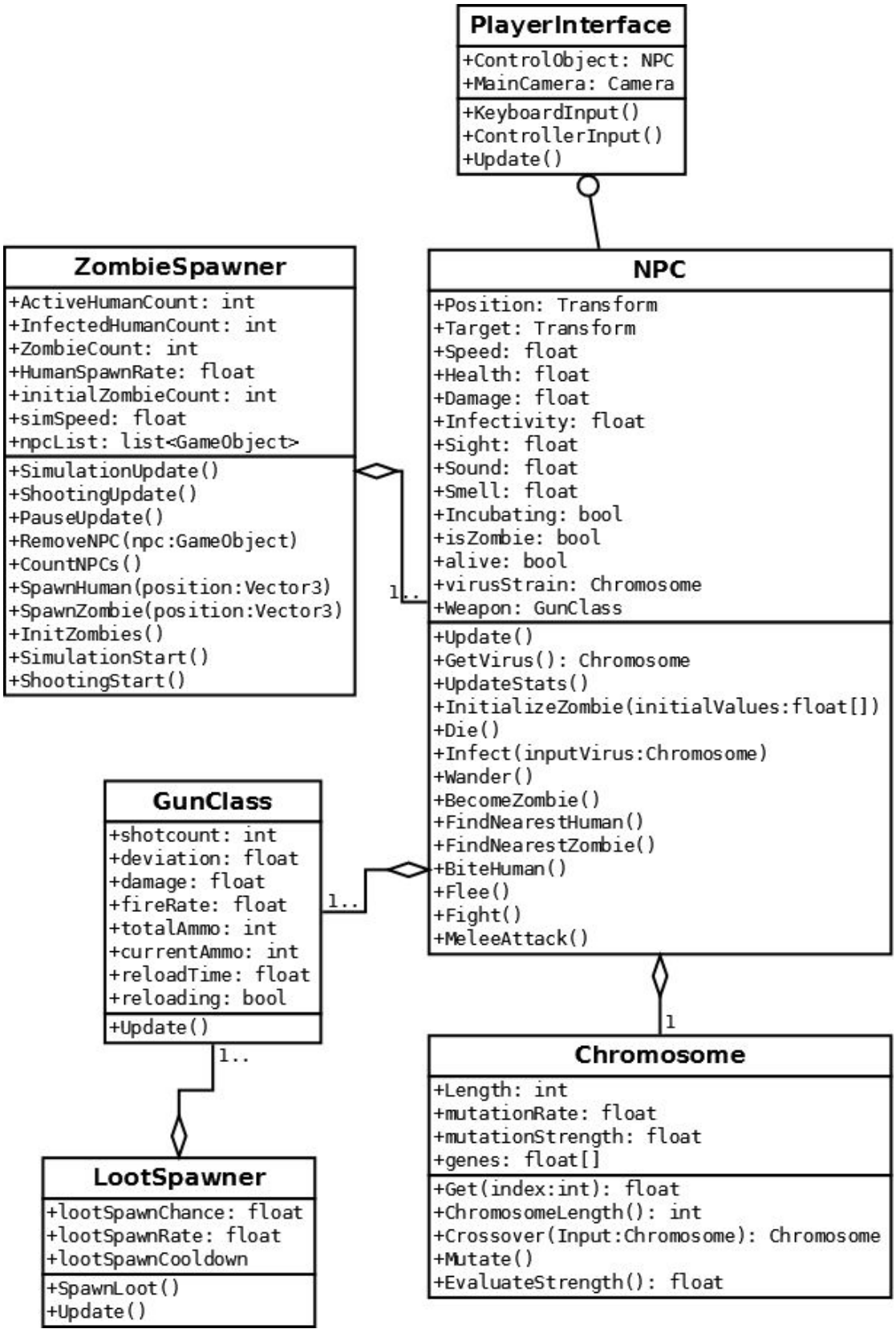
Interface for user input that can be applied to any NPC, allowing for player control to transfer when they current player is infected/dies.

Class interaction

Game state flow diagram:



Next page - UML Diagram:


PlayerInterface
+ControlObject: NPC
+MainCamera: Camera
+KeyboardInput()
+ControllerInput()
+Update()
ZombieSpawner
+ActiveHumanCount: int
+InfectedHumanCount: int
+ZombieCount: int
+HumanSpawnRate: float
+initialZombieCount: int
+simSpeed: float
+npcList: list<GameObject>
+SimulationUpdate()
+ShootingUpdate()
+PauseUpdate()
+RemoveNPC(npc:GameObject)
+CountNPCs()
+SpawnHuman(position:Vector3)
+SpawnZombie(position:Vector3)
+InitZombies()
+SimulationStart()
+ShootingStart()
1..
NPC
+Position: Transform
+Target: Transform
+Speed: float
+Health: float
+Damage: float
+Infectivity: float
+Sight: float
+Sound: float
+Smell: float
+Incubating: bool
+isZombie: bool
+alive: bool
+virusStrain: Chromosome
+Weapon: GunClass
+Update()
+GetVirus(): Chromosome
+UpdateStats()
+InitializeZombie(initialValues:float[])
+Die()
+Infect(inputVirus:Chromosome)
+Wander()
+BecomeZombie()
+FindNearestHuman()
+FindNearestZombie()
+BiteHuman()
+Flee()
+Fight()
+MeleeAttack()
GunClass
+shotcount: int
+deviation: float
+damage: float
+fireRate: float
+totalAmmo: int
+currentAmmo: int
+reloadTime: float
+reloading: bool
+Update()
1..
1..
1
Chromosome
+Length: int
+mutationRate: float
+mutationStrength: float
+genes: float[]
+Get(index:int): float
+ChromosomeLength(): int
+Crossover(Input:Chromosome): Chromosome
+Mutate()
+EvaluateStrength(): float
LootSpawner
+lootSpawnChance: float
+lootSpawnRate: float
+lootSpawnCooldown
+SpawnLoot()
+Update()

Level details
City is open for exploration, but bordered by water on all sides, limiting player movement outside of the playable area. It will be one continuous region without loading screens and divided into several distinct areas defined by their building types: residential, business district, docks, and hospital. Should performance issues arise due to the stress of simulating such a large number of NPC's, physical enemies can be replaced by a simple genetic algorithm or by stripped-down versions of the NPC class that contain only AI, position, and viral strain elements when the user is not in close proximity.
Some buildings will have areas that can be entered, such as some garages and the interior of the hospital.

Menu details
The main menu will consist of several buttons each linking the player to other menus or gamestates:

- The 'Survival mode' button will place the player into the survival gamemode (one zombie spreading infection, survive for as long as possible)
- The 'Simulated mode' will take the player into the simulated gamemode (simulates game for a certain amount of time or until certain amount of zombies is reached before dropping the player in)
- The 'Options' button will link to the options menu that will allow the player to change certain aspects of the game (audio on/off, etc)
- 'Exit' button will close the game

All menu interactions will be done via mouse input. The mouse cursor will be visible and button will highlight when hovered over. Buttons will provide an audio response when clicked so that the player knows that they have interacted with the button successfully.

The main menu will show the game title and have in game screenshots shown faded in the background, as to not distract from the more important elements on the menu, ie buttons.

Art Requirements
Strain will require multiple art assets implemented in the game:

- Zombie character models (including variations of the regular, non-evolved zombie, and other zombie types like the hulking zombie) including multiple animations each for different actions (walking, running, biting, death)
- Player character model complete with animations
- Regular human model (multiple variations)
- Weapon and ammunition models
- Environment assets (houses, lamps, roads, cars, etc)
- UI elements (health bar, stamina, remaining ammo, etc)
- Feedback effects (blood effects on screen after taking damage, bullet flashes)

Sound Requirements

Strain will require multiple sound assets implemented in game:

- Gun sounds (firing gun, reloading gun, switching weapon, empty magazine) different for each gun type
- Melee attack sounds (attacking, different sound for when an attack lands or misses)
- Walking sounds (for humans, zombies and player)
- Dashing sounds (for player)
- Getting hurt/taking damage sounds (for humans, zombies and player)
- Random zombie sounds (moans, roars, attacking sounds, death sounds)
- Random human sounds (talking, screaming, crying)
- Sounds to indicate that a human is turning into a zombie (a unique death scream or squelching sound)
- Environment sounds (background noise like bullets or faraway screams)
- Background music (varying in intensity based on what is happening on screen, e.g. intense music when a horde of zombie enters the screen)
- Radio sounds (equip radio, put away radio, interact with radio)
- Menu sounds (click button, hover button, menu background music)