# Technical Design Document

## Working Title: Horde

### Horde Team

#### Reece Howe

Contents

[Technical Design Document 1](#_Toc458070784)

[Working Title: Horde 1](#_Toc458070785)

[Horde Team 1](#_Toc458070786)

[Introduction 4](#_Toc458070787)

[Swarming the City Overview 4](#_Toc458070788)

[Quarantine Zone 4](#_Toc458070789)

[Territories 4](#_Toc458070790)

[Safe Zones 5](#_Toc458070791)

[Horde Size (Stretch Goal) 5](#_Toc458070792)

[Territory Reclamation (Stretch Goal) 5](#_Toc458070793)

[Monetising (Stretch Goal) 5](#_Toc458070794)

[Controller Scheme 6](#_Toc458070795)

[Menu Navigation 7](#_Toc458070796)

[Stretch Goal: 3D Interactive Menu 7](#_Toc458070797)

[Stretch Goals: Special Effects 7](#_Toc458070798)

[Key Systems 8](#_Toc458070799)

[Game Controller 8](#_Toc458070800)

[Menu Controller 8](#_Toc458070801)

[HUD 8](#_Toc458070802)

[Navigation Mesh 9](#_Toc458070803)

[Zombie 9](#_Toc458070804)

[Human 9](#_Toc458070805)

[Obstacle Mechanics 9](#_Toc458070806)

[Stretch Goal: Vehicles 10](#_Toc458070807)

[Animation List 10](#_Toc458070808)

[Sound List 10](#_Toc458070809)

[Class Diagrams (NOT REQ DMC) 11](#_Toc458070810)

[UML (NOT NEEDED DMC) 11](#_Toc458070811)

[Menu Navigation 12](#_Toc458070812)

[Animations (NOT REQ DMC) 13](#_Toc458070813)

[Program Flow 14](#_Toc458070814)

[Platform Considerations 15](#_Toc458070815)

[Deployment Platforms 15](#_Toc458070816)

[Imported Assets 15](#_Toc458070817)

[Required Tools – Game Dev 15](#_Toc458070818)

[Required Tool – Project Management 15](#_Toc458070819)

[Organisation and Reference 16](#_Toc458070820)

[Reference Documents: 16](#_Toc458070821)

[Filename Conventions 16](#_Toc458070822)

[Folder Structure 16](#_Toc458070823)

[Art Detail 16](#_Toc458070824)

[Level Design 17](#_Toc458070825)

[Unity Specific 18](#_Toc458070826)

[Contributors 19](#_Toc458070827)

[Role and Responsibilities 19](#_Toc458070828)

[Conclusion 20](#_Toc458070829)

[Appendices 21](#_Toc458070830)

[Appendix A: Introductory Tutorial Level – “Escape the science Facility” 21](#_Toc458070831)

[Appendix B: Asset List 21](#_Toc458070832)

## Introduction

“Shepherd your Undead Horde!”

Horde is an Isometric strategy game for mobile phones.

The player swipes across the horde shepherding them toward their goals. Attack the living, grow your horde, and defeat the strongholds, escape the quarantine.

As your horde spreads through the city humans will created defenses and attempt to take out your horde. Devour the fallen to recover horde health, take care not to pick the bones cleans otherwise the fallen will not rise again.

The zombie virus has varying effects on its hosts. You may find zombies with a variety of abilities to help you on your way.

The goal of this prototype if to produce a Quarantine Zone with seven Territories (Levels).

Stretch Goals Include; a tutorial level, zombie bar, research facility, control center, and 3d interactive menu.

## Swarming the City Overview

Navigate through the city, infect humans and grow your horde. Difficulty level rises as your horde spread through the territories. The City is divided in to quarantine zones which consists of smaller territories (levels). For the purpose of this project we will be creating one quarantine zone.

### Quarantine Zone

Consists of multiple Territories, which the player will need to conquer with their Horde. Conquer all territories to break out of a Quarantine Zone. Completing a Quarantine Zone will unlock paths to new Quarantine zones. These zones may be neighbouring areas, alternatively they could be connected by a transport hub; planes, train, automobiles, and boats, etc. Quarantine zones have unique aesthetic and pseudo story.

### Territories

Each territory represents a level which the player will need to overcome. A territory is an urban area with 3 or more Safe Zones which will need to be overcome before the player can escape the Territory. Overcoming a territory unlocks neighbouring territories.

### Safe Zones

Each safe zone contains a number of humans who will defend the safe zone. Safe Zones will become progressively harder to overcome as you progress through the game. These zones will be indicated by to the player via the UI so the player will know where the next objective is. Safe Zone may have automated defenses. The player may need a minimum number of zombies in order to break through barricades\safe zone gates.

### Horde Size (Stretch Goal)

Between Levels a player’s zombie horde will replenish over time. The maximum horde size is determined by the number of territories a player controls. Territories may have a recommended starting zombie count. “Your Horde must be this big to enter.” Zombie you finish a level with a returned to your zombie pool.

### Territory Reclamation (Stretch Goal)

After players are away for a certain amount of time territories can be taken back by humans. Players receive notifications when their territories are under attack. Quicker response results in less safe zones in the territory being taken by the human.

### Monetising (Stretch Goal)

Watch an ad to refill your zombie pool. Post Release Quarantine Zones could be released as (Expansion, update, or DLC)

### Controller Scheme

#### Gesture Controls

Tap Controls



Tapping the map will move the camera to the selected location

Tapping a UI button will activate the button.

Swipe Controls



Swiping over Zombies will cause them to move in the direction of the swipe.

The effectiveness of you swipe is changes based on the number of zombies affected by the swipe.

Multiple swipes over the same zombie increases the speed momentarily.

Stretch Goal: Influence range the radius of the unit selection for a swipe.

Stretch Goal: Tap and Hold to set the camera to follow a zombie

#### Horde Properties

The following is a list of things which will need to be considered during movement of the horde:

* Number of Zombies
* Calculate acceleration (decrease with more zombies). Having more zombies makes the horde less maneuverable.
* Calculate Max Speed
* Calculate Desired Velocity
* Calculate Noise Generated (increase with more zombies) (alert humans and attracts zombies)
* Momentum: increasing momentum increases the zombie damage and noise.

### Menu Navigation

Prototype Navigation

R Restarts the Level

Escape Quits the game

### Stretch Goal: 3D Interactive Menu

Control Room for Territory Selection and Overview of progress

Zombie Bar for a representation of the types of zombies you might have discovered.

Laboratory for researching Horde Upgrades.

### Stretch Goals: Special Effects

Gore

Muzzle Flash

Projectile Ray Cast

Exhaust

Fog of War

## Key Systems

### Game Controller

There will be a single GameController GameObject in the Territory (level) with the following scripts attached:

UserInputManager: collects raw swipe data and sends it to the appropriate system

ObjectManager: owns master lists of zombies, humans and abnormals

ObjectiveManager: Keeps track of in game objectives, like the number of safe zone defeated

SoundLibrary: Store a list of all the sounds for the game, so they can be called by name

SceneManager: Manage loading of scenes

NoiseManager: Creates hot and cold zones for noise on the map

#### Stretch Goals:

ConvoyManager: Managing supply vehicle routes between safe zones.

StatisticsManager: Managing in stats

AchievementManager: Manages Achievement, like speed runs, etc.

TerritoryReclamationManager: determines whether captured territories have return to human control

FogOfWarManager: keeps track of explored areas, and areas currently in sight

### Menu Controller

There will be a single MenuController GameObject in the TerritorySelector Scene with the following scripts attached:

SceneManager: Manage loading of scenes

QuaratineManager: Keeps tracks of completed territories

### HUD

There will be a HUD GameObject in the Territory (level) Scene with the following scripts attached

HordeProperties: keeps track of zombie related information, like count

IndicatorManager: managing Objective and Noise Indicators

ObjectiveManager: keeping track of captured safe zones

AbnormalManager: keep track of available abnormals

Stretch Goal: Mission Timer

### Navigation Mesh

Agents in the game will be utilizing the unity’s in game Navigation Mesh

### Zombie

There will be a Zombie GameObject with the following scripts:

Movement: for managing zombie destination and speed

Attack: simple attacks and devouring of humans when the living come in range

Health: keeps track of health , damage, and effects

InfectionStatus: keep track if the infections status of our zombie

CleanUp: returns zombies to the zombie pool when no longer needed

Behaviour: this will manage how the zombies reacts to different input like noise levels and nearby humans, and other input.

Stretch Goal: Special Abilities: some zombies will have special abilities this scrip will manage these effects.

### Human

There will be a Human GameObject with the following scripts

Behaviour: this will manage how the zombies reacts to different input like noise levels and nearby zombies, and other input. Behaviours will include; Flee, Cower, Melee / Ranged Attack, Patrol, Guard, and Death.

Awareness: Checks for incoming threats, like zombies, noise

InfectionStatus: keep track if the infections status of our zombie

Reanimation: manages replacement of infected dead humans with zombies utilizing the zombie pool

Stretch Goal: Tactics (Squad Goals): Humans uses tactics and traps to lure zombies and kill them

Stretch Goal: Classes: There is a variety of classes which change the way humans behave.

### Obstacle Mechanics

Building which are destructible like; Barricades, Guard towers, etc.

### Stretch Goal: Vehicles

NPC Vehicles which travel through the streets

### Animation List

* Human
* Zombie
* Barricade
* Environmental

### Sound List

* Zombie
* Human
* Environment
* UI
* Gun Shot
* Barricade Destruction

## Class Diagrams (NOT REQ DMC)

## UML (NOT NEEDED DMC)

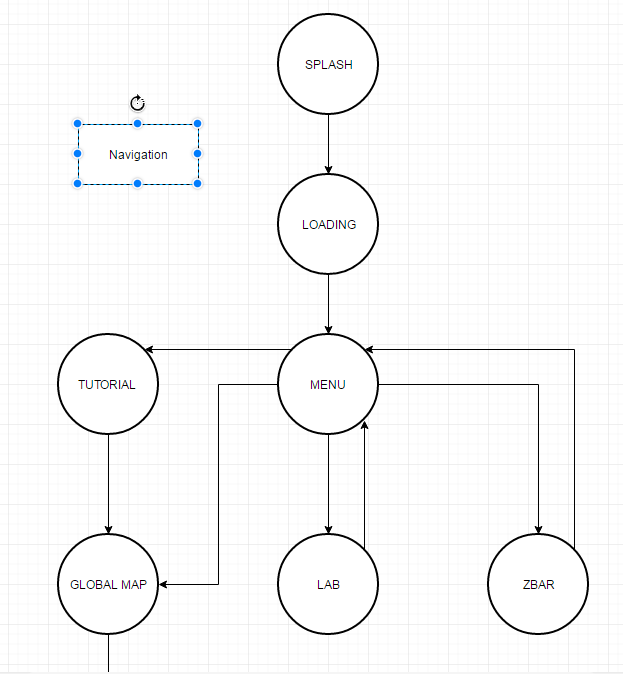
Behaviours

Abnormals – TBD

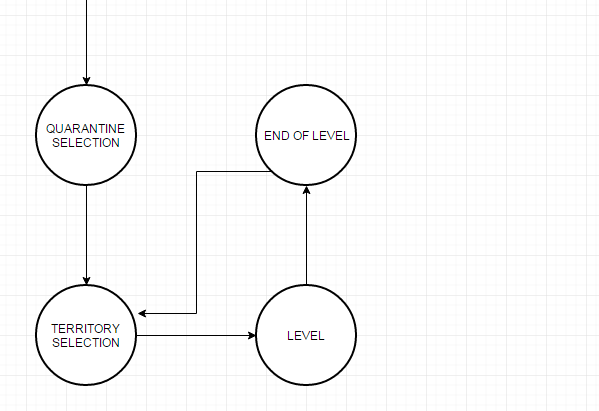
Human Classes - TBD

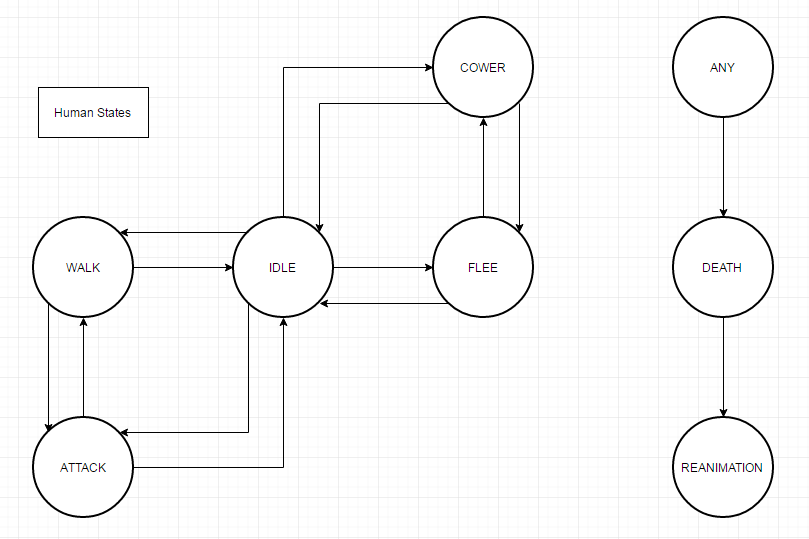
### Menu Navigation

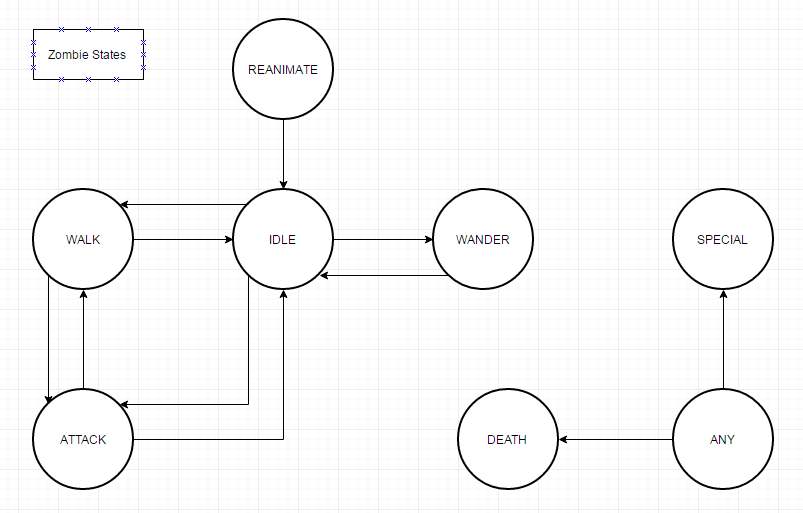
Stretch Goals



Technical Demo



Animations (NOT REQ DMC)



## Program Flow

Splash

Territory Selector Scene

Load Level Scenes

If Win Load Territory Selector

If Lose Load Territory Selector

## Platform Considerations

### Deployment Platforms

Android

Stretch Goal: IOS

16x9 screen ratio

### Imported Assets

None

### Required Tools – Game Dev

Microsoft Visual Studio 2015

Unity3D 5+

Maya

Photoshop

### Required Tool – Project Management

GitHub and Source Tree – version control

Trello – task management

Slack Team – team discussion

Facebook – team event management

Google Docs – class diagrams

https://www.draw.io/ for Flowchart, Class Diagrams, and UML

## Organisation and Reference

Here is a recommended list of organizational information.

### Reference Documents:

Unity Learn:

<http://unity3d.com/learn>

Unity Best Practices:

<http://devmag.org.za/2012/07/12/50-tips-for-working-with-unity-best-practices/>

GitHub:

https://github.com/NotProGamer/Horde

### Filename Conventions

All Source code files will be capitalized with meaningful names (eg. Health.cs)

Game Assets to be named using Title case much like source code

AIE Coding standards to be used in source code creation

### Folder Structure

All Imported Assets to be left in the root directory.

All Team created assets to be placed in subdirectory with similar structure.

Example:

*Assets folder*

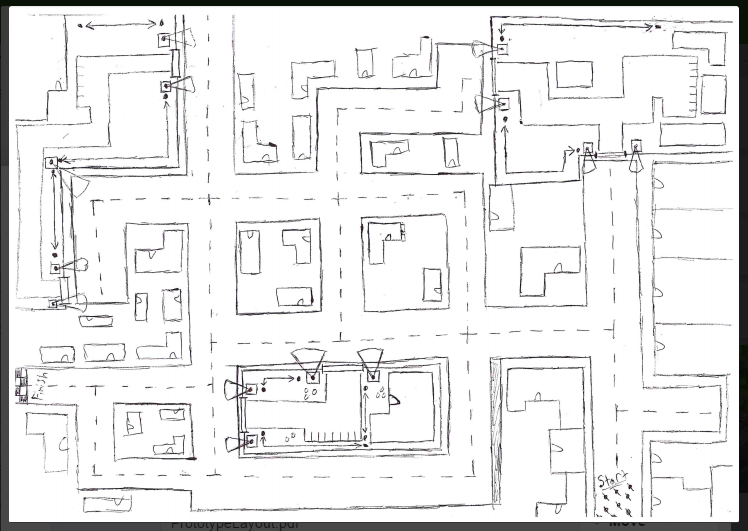
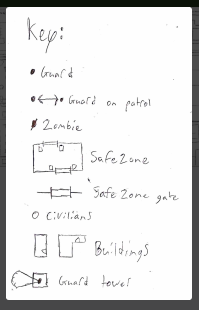
* *Project*
  + *Scripts*
  + *Prefabs*
  + *Etc.*
* *Standard Assets*
  + *Etc.*

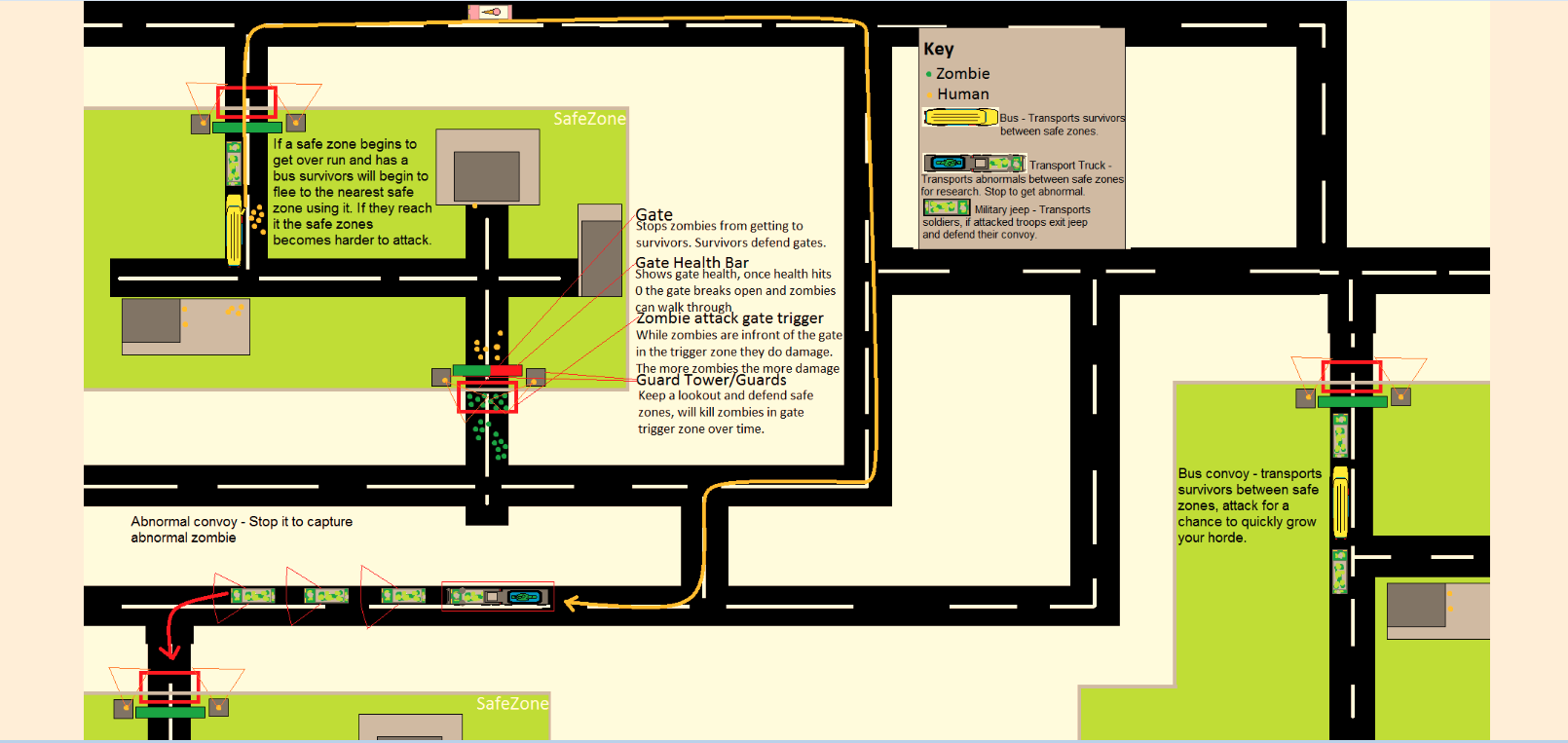
### Art Detail

Scale to be determined and adhered to in order to directly import assets in to the game.

### Level Design

Assets to be designed created by our designers Rory and Brandon.



### Unity Specific

**Tags**

Player

Zombie

ZombieLure

GameController

Barricade

Human

Building

GuardTower

Guard

**Tag Qualifiers**

bool IsZombie(GameObject other) // check if a game object has one of the zombie tags

bool IsHuman(GameObject other) // checks if a game object is a human

bool IsDestructible(GameObject other) // check if an object is destructible

bool CanBeInfected(GameObject other) // check if an object can be infected

bool IsDevourable(GameObject other) // check if an object is devourable

**Layers**

Ground

## Contributors

Rory Caire-Mccaul

Brandon Nicholas

Adrianna White

Aanikka Holder

Reece Howe

### Role and Responsibilities

Project Manager – Rory

Prefab Manager - Brandon

Programmer – Reece

Environmental Artist – Addie

Character Artist and Animator – Aanikka

UI Design – Rory

UI Artist – Brandon

Story – Rory

Source Control Conflict Manager – Reece

QA Testers

Sound Technician

Deployment and Distribution

## Conclusion

*Prototype Complete. \*Drops Mic\**

Thanks to **Academy of Interactive Entertainment** for helping our team get this prototype finished.

Special Thanks go to our lectures …. for all their help and support during the making of this prototype.

Super Special Thanks to ***Rory, Brandon, Adrianna, Aanikka, Joanna, Zac***

## Appendices

### Appendix A: Introductory Tutorial Level – “Escape the science Facility”

Control a newly reanimated zombie. Infect scientists, workers, and guards growing your horde. Gather enough zombie to take over the facility and break free of a government lab, kicking off the apocalypse.

This would be a mini game that introduces the player to the mechanics of the game.

### Appendix B: Asset List

#### Character Art

Character Assets

* Zombies
  + Z
* Humans
  + H
* Abnormals
  + A

#### Environmental Art

Environmental Assets

Props

* Foliage
  + Prop\_PotPlant\_Mod\_01.fbx
  + Prop\_PotPlantDouble\_Mod\_01.fbx
  + Prop\_Tree\_Mod\_01.fbx
* Fences
  + Fence\_BwStaight\_Mod\_01.fbx
  + Fence\_BwCorner\_Mod\_01.fbx
  + Fence\_ChainStaight\_Mod\_01.fbx
  + Fence\_ChainCorner\_Mod\_01.fbx
* Gates
  + Gate\_FancyLeft\_Mod\_01.fbx
  + Gate\_FancyRight\_Mod\_01.fbx
  + Gate\_BwLeft\_Mod\_01.fbx
  + Gate\_BwRight\_Mod\_01.fbx
  + Gate\_ChainLeft\_Mod\_01.fbx
  + Gate\_ChainRight\_Mod\_01.fbx
* Objects
  + Prop\_Trashcan\_Mod\_01.fbx
  + Prop\_SodaCup\_Mod\_01.fbx
  + Prop\_CafeTable\_Mod\_01.fbx
  + Prop\_CafeChair\_Mod\_01.fbx
* Doors
  + Door\_GenericLeft\_Mod\_01.fbx
  + Door\_GenericRight\_Mod\_01.fbx
* Cars
  + Prop\_Car01\_Mod\_01.fbx

Roads

* Asphalt roads
  + asphalt-road\_T-section.tga
  + asphalt-road\_Straight.tga
  + asphalt-road\_Cross.tga
  + asphalt-road\_Corner.tga

Buildings

* Generic Building models
  + Building\_Generic\_Mod\_01.fbx
  + Building\_Generic\_Mod\_02.fbx
  + Building\_Generic\_Mod\_03.fbx
  + Building\_Generic\_Mod\_04.fbx
  + Building\_Cafe\_Mod\_01.fbx
  + Building\_Cafe\_Mod\_02.fbx
  + Building\_SodaShop\_Mod\_01.fbx

#### Animations

Animations will be required:

* Characters
  + Anim\_Human\_Idle\_001.fbx
  + Anim\_Human\_Walk\_001.fbx
  + Anim\_Zombie\_Idle\_001.fbx
  + Anim\_Zombie\_Walk\_001.fbx
* Gates
  + Anim\_FancyGateOpen\_01.fbx
  + Anim\_BwGateOpen\_01.fbx
  + Anim\_ChainGateOpen\_01.fbx
  + Anim\_FancyGateClose\_01.fbx
  + Anim\_BwGateClose\_01.fbx
  + Anim\_ChainGateClose\_01.fbx
  + Anim\_FancyGateBreak\_01.fbx
  + Anim\_BwGateBreak\_01.fbx
  + Anim\_ChainGateBreak\_01.fbx

#### Sound Effects

Sound Effect will be required:

* Zombies
  + SFX\_Moaning\_01
  + SFX\_Attack\_01
  + SFX\_Footsteps\_01
  + SFX\_Dying\_01
  + SFX\_Reanimating\_01
  + SFX\_Eating\_01
* Humans
  + SFX\_Screaming\_01
  + SFX\_Shooting\_01
  + SFX\_Footsteps\_01
  + SFX\_Dying\_01
  + SFX\_Punching\_01
  + SFX\_Alarmed\_01
* Misc
  + SFX\_Explosion\_01
  + SFX\_CarAlarm\_01
  + SFX\_PoliceSiren\_01
  + SFX\_GateBreak\_01