

WELCOME TO H.E.L.L.

(The Healing Emporium for Lost Loveables)



Centaur - The Team

- Anna Tito - Designer - 3195610
- Alex Dircks - Programmer - 3190954
- Boden Tennent - Artist - 3197693
- Gareth Campbell - Programmer - 3191338
- John Stavropoulos - Programmer - 3191010
- Lucas Hepner - Artist - 3166343

What we did:

Whole Team:

- Brainstorm character concepts and features
- Brain storm weapon and item concepts
- Plan game style - rendering, use of camera view & necessary AI

Individuals:

Anna Tito

- Joint team management and project co-ordination.
- Preliminary Mod-Development: Story and Character development, Logo design, Level Floor Plans.
- Design Documentation: Design document, Art Bible and Technical Bible design and layout, Design Document & Art Bible Content, CD & CD case design.
- Concept Art: Mr Bunnikins, Snoopy, Lucky & full body Toymaker Sketches.
- WTH Mod: All levels - Textures, extra meshes, movers, placing bots, botpaths, placables, pick up bases, volumes, particle effects, re:texture meshes, re:lighting, level geometry and mesh

refinement, Level design - WTHCastle_lvl16, WTHLibrary_lvl5.

- Textures: Comic Texture Creation for levels, weapons (animations and meshes, inc ammo).
- Modeling: Bookshelves, desk (with decorations), desk trays, pencil holder, chairs, pick-up bases, castle (ramparts and castle), sniper ammo, cell-door, grate, Gates of Hell.
- Mod Graphics: HUD Graphic, Comic strip Art-Snoopy, The Toymaker, Comicstrip creation (All)
- Presentations: 1 content & final layout, 2 & 3.
- Research: Sound effects, movers, volume application.
- QA: Continuous level testing, bug fixing and level flow tweaking.

John Stavropoulos

- Joint project planning and team management
- Technical Design Document: List of scripts, List of classes, justifications
- WTH Mod: Directory structure, Default.ini, Defuser.ini, UT2k4mod.ini, .upl file, .int file, added OSMT package to our mod
- Code: Created all character classes, all weapon classes including custom fire and projectile for pumpkin flamer, gametype, HUD, mutator, main menu, player class
- Modelling: Snoopy, flippy, woodstock, egg ammo, paint ammo
- Exporting/Importing: Exported/Imported all characters and all weapons meshes and static meshes from Maya to UT, also exp/imp some other static meshes
- Textures: Created font package, fixed alpha and implemented all HUD textures and placement,

created textures for snoopy and woodstock, uv maps for flippy and toymaker, imported character textures for flippy, teddys, toymaker and bunnikins

- Skin/Rig: Used the Male Skeleton on UDN to rig all characters (teddy, flippy, snoopy, bunnikins, toy maker)
- Research: Using maya for modelling, osmt, actorX plugin, weapon placement and structure, character/enemy model skeleton structure
- Levels: Level 1 Dungeon BSP, transitions between levels with boss checks, setup of triggered player starts
- Music: Converted custom music tracks to .ogg and added them to the mod
- QA: Continuous weapon/character testing

Boden Tennent

- Character modeling: Toymaker, First version of Mr Bunnikins.
- Weapon modeling: Paintgun, Flamethrower, Egg Gun, Woodstock Cannon, Vacuum-cleaner.
- Mod Graphics: Comic strip Art - Lucky.
- WTH Mod: WTHkitchen&garden_lvl4 - Level Design, Meshes, lighting, Movers & textures (version 1), WTHdungeon_lvl1 - Meshes, lighting & textures (version 1).
- Concept Art: Teddy bears and other characters, weapons, levels, towers, design document teddy graphic.
- Character textures: Flippy, Toymaker.
- Presentations: 1 initial layout.
- Static Meshes: Door frame, door, double door frame, health kit, Cell door, Hospital door (x2).
- Sketching and Modeling of Vacuum Gun
- Textures for models (not yet implemented).

Lucas Hepner

- Concept Art: Concept art for characters, pickups etc.
- WTH Mod: WTHwards_lvl2- Level design, lighting & texturing (version 1), WTHwards_lvl3- Texturing (version 1), Version 1 of WTH Textures
- Research: Model importing
- Music: Outsourcing for level sound tracks
- Character modelling: Teddy model, Mr. Bunnikins final.
- Mod Graphics: Comicstrip Mr Bunnikins.
- Character Textures: Mr Bunnikins & Teddies.

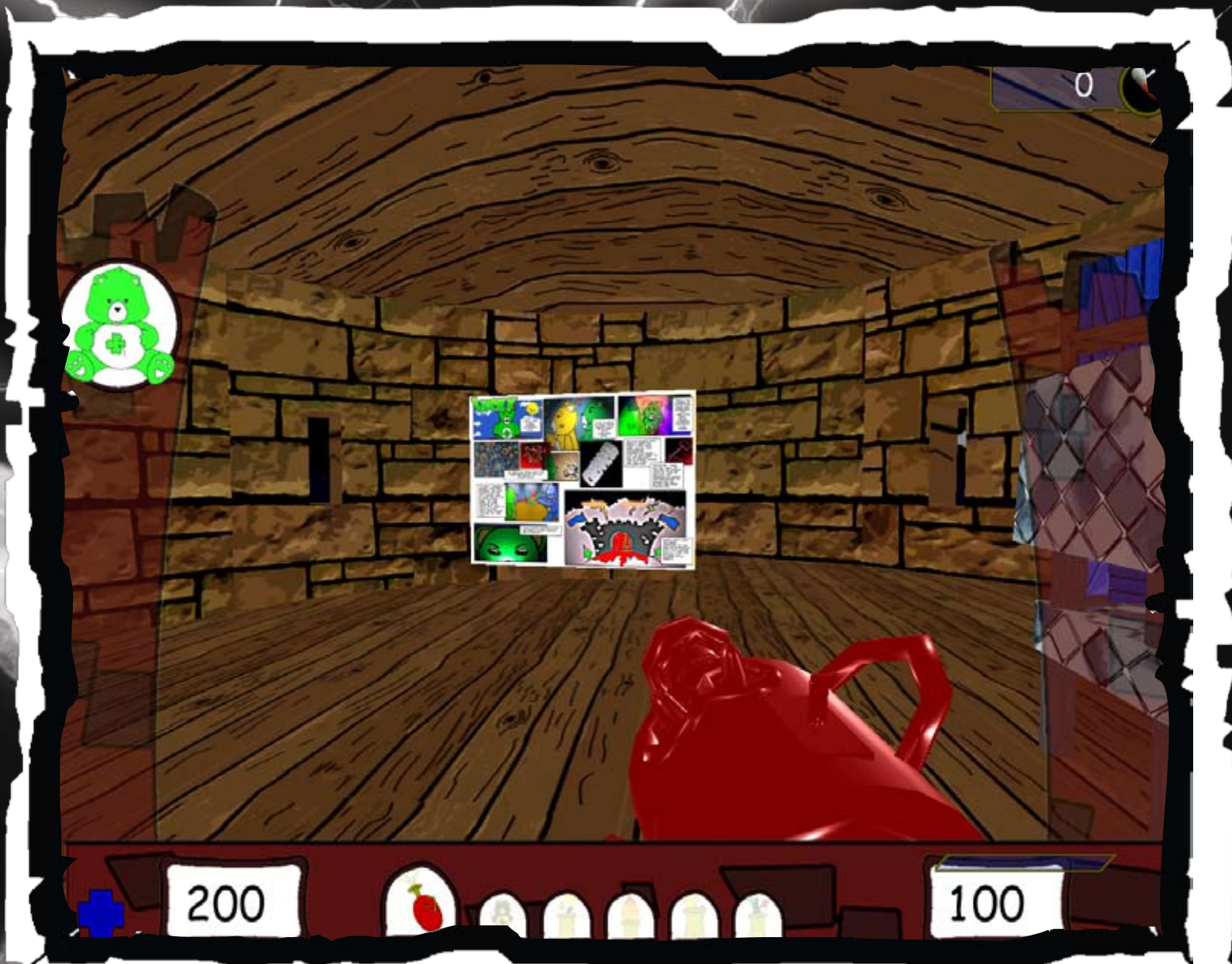
Alex Dirks

- Research: HUD, mesh importing
- Preliminary Mod-development: Rough character sketches & concepts, HUD style, weapon concepts
- WTH Mod: WTHwards_lvl3 - Level design, lighting & meshes (version 1), WTHwards_lvl2 - Meshes & texturing (version 1)
- Meshes: Decorative Meshes

Gareth Campbell

- Preliminary Mod-development: General castle map, Basic game flow and progression
- Research: Mover application research, OSMT Bot behavior, Getting to work the patrol points and bot patrolling, Finding a way to make items only able to be picked up by the player.
- Code: Customised health pickups
- Testing: OSMT bot extending, a teleport/trigger combo so would only teleport once area cleared. (Incomplete)

Custom music created for WTH by Kym Dillon



You wake up in H.E.L.L.



Contents

Centaur - The Team.....	1
Overview.....	5
Game Mechanics	6
Interactions	6
World	6
Characters	6
NPCs	6
H.E.L.L. Castle Layout.....	7
Game Progression.....	10
Characters	12
Good Luck Bear a.k.a Lucky.....	12
The Toymaker	13
Mr Bunnikins	13
Snoopy	14
NPCs	15
AI.....	16
Game Elements.....	17
Items.....	17
Weapons and Ammo	17
References:	19
Art Bible	21-46
Script Bible.....	47-60
Final Addendum	61-64





Welcome to HELL

(Healing Emporium for Lost Lovables)

Once upon a time there was a forest and in this forest there was a hill and on this hill stood a castle, the most frightening castle you have ever seen. It was surrounded by a pointed fence wound about with brambles and over its gates read a sign, 'Abandon hope all ye who enter here.' This is the place the bad toys go. If you look closer, on the top floor, there is a window, furthest from the gate, where a small light flickers. In this room sits a Care Bear, once known as Good Luck Bear, now simply known as lucky. He sits in the flickering gloom sharpening a butter knife to a deadly edge.....

Overview

Welcome to the Design Bible for 'Welcome to H.E.L.L.' (WTH), this document aims to give you an introduction to the world and game play of WTH.

WTH is set in an alternate reality where the toys are sentient beings. You are in H.E.L.L a castle that has been converted into a psychiatric hospital for toys who have been deemed criminally insane. You awake in the middle of a mass break out orchestrated by none other than the great Toy Master himself, it is time for you to be free.

Using a combination of game play and comic strip interludes WTH takes you through the protagonist's story and those of some of the other inmates. You must progress through the darkest layers of H.E.L.L. before you can reach the outer limits and a fugitive's tenuous freedom.

Game Mechanics

WTH is a horror-comedy adventure first person shooter (FPS). We are designing this game to use the standard UT joint keyboard and mouse game controls. In terms of camera control we will be designing this game to use the first person camera view though hopefully we will be able to expand that to third if we have time.

We are planning to use comic strip interludes to progress the plot and give an introduction to the back story of the major characters you will fight. This will allow us to focus on the action in the playable areas while still advancing the plot in an interesting way.

Interactions

World

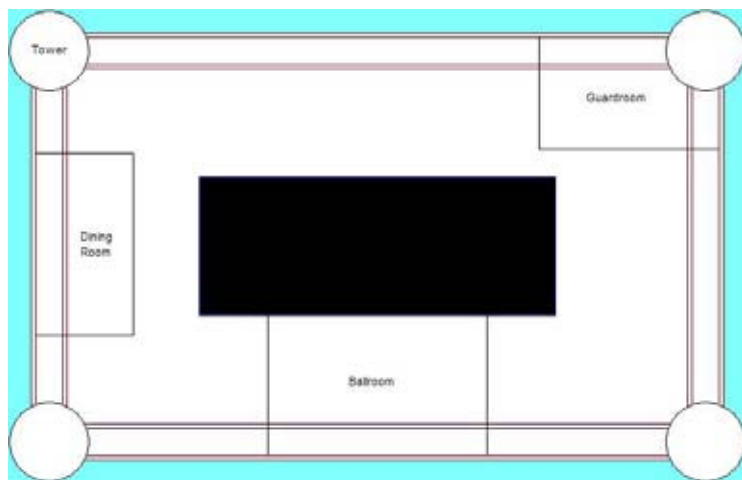
The world is fairly standard, since the game is set in a converted castle, most of the interactions in the game are fairly low tech. Level progression will be through doors or via stairs.

Characters

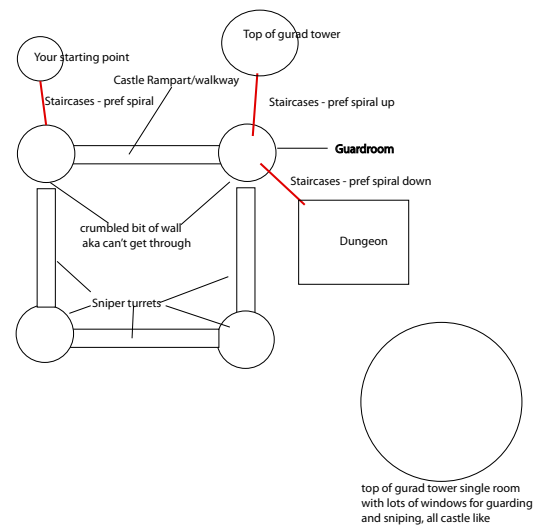
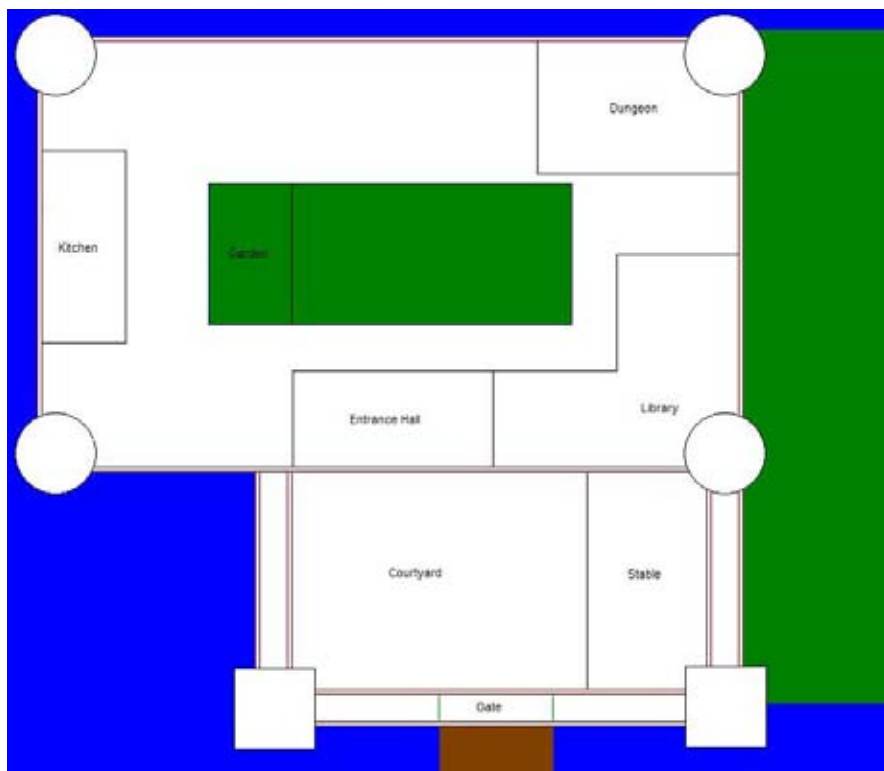
The developed characters in this mod are the protagonist and our various bosses, the interactions with each of them are hostile. However the comic strip interludes allow for character and story development. This method however keeps our story and the character interactions very controlled and linear.

NPCs

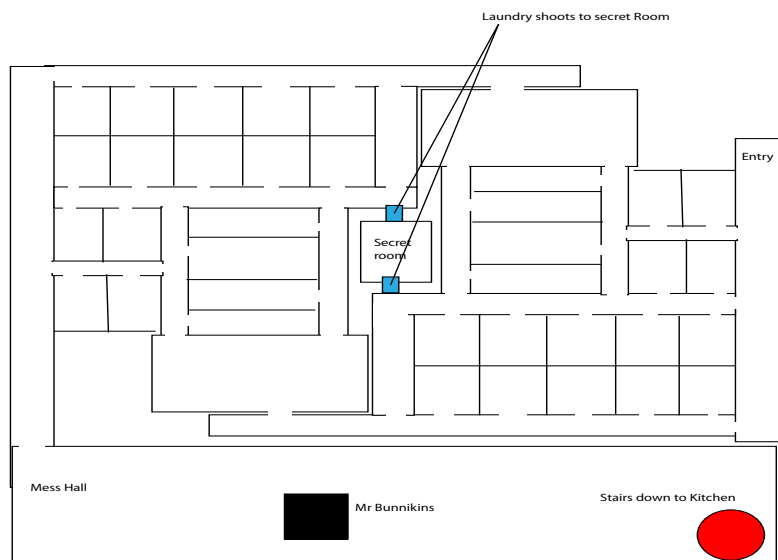
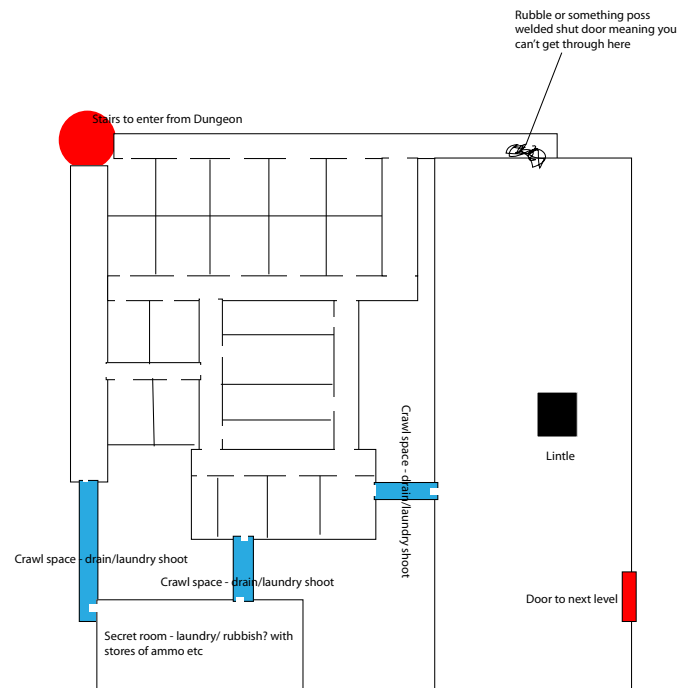
We will set up the NPCs not to be hostile to each other; this limits the amount of chaos that can be created with kamikaze and melee characters. Many of the NPCs have specific interaction modes, those that use a Kamikaze type attack, will use a close proximity melee, so that they can get the best shot at you. Since it is based in an asylum most of the generic characters will end up using a melee attack. Only some of the Bosses the guards and some of the teddies will actually use weapons.



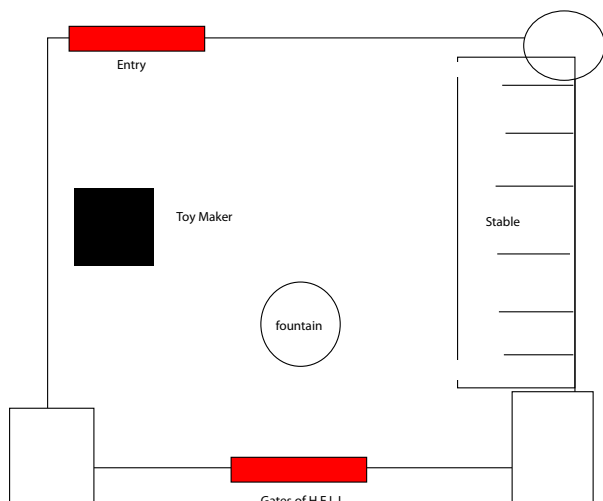
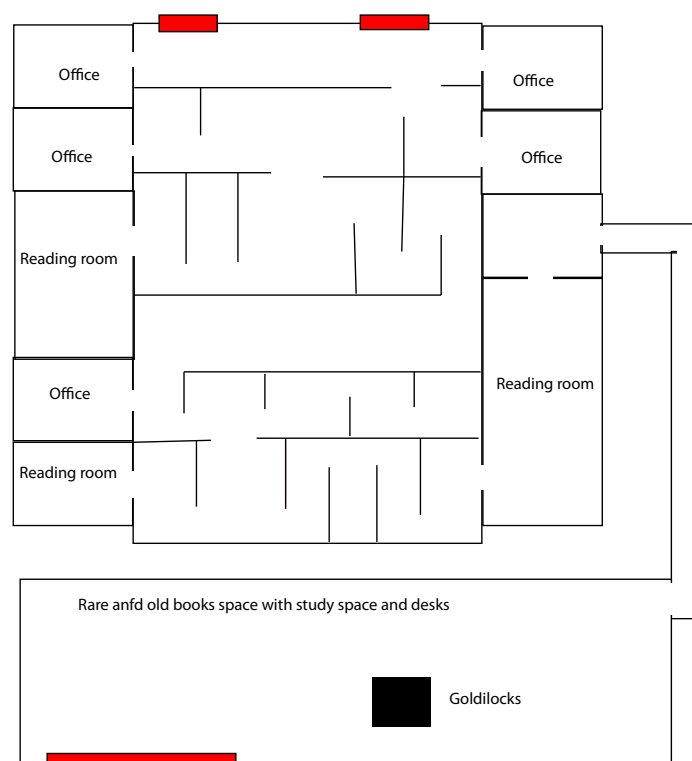
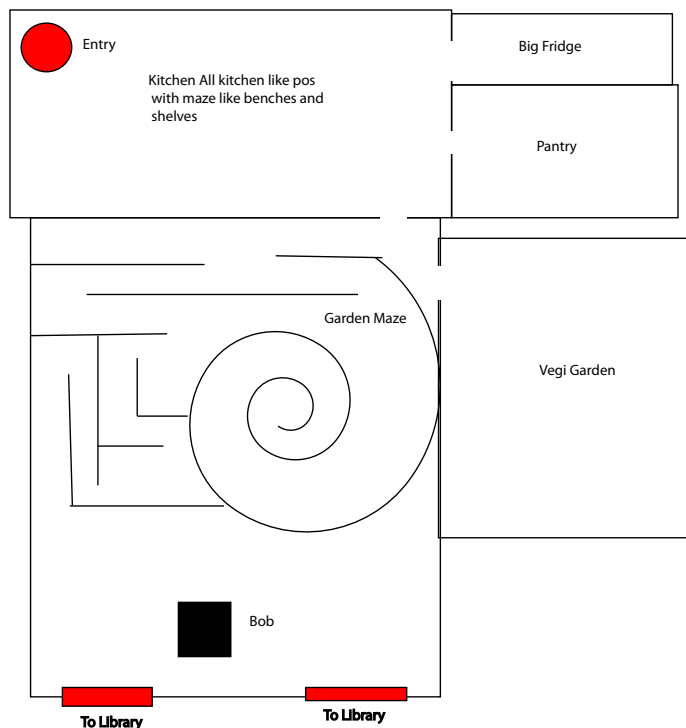
H.E.L.L. Castle Layout



From top left: Castle top floor, Castle Ground Floor, Castle Ramparts & Guard rooms (Level 1.1)

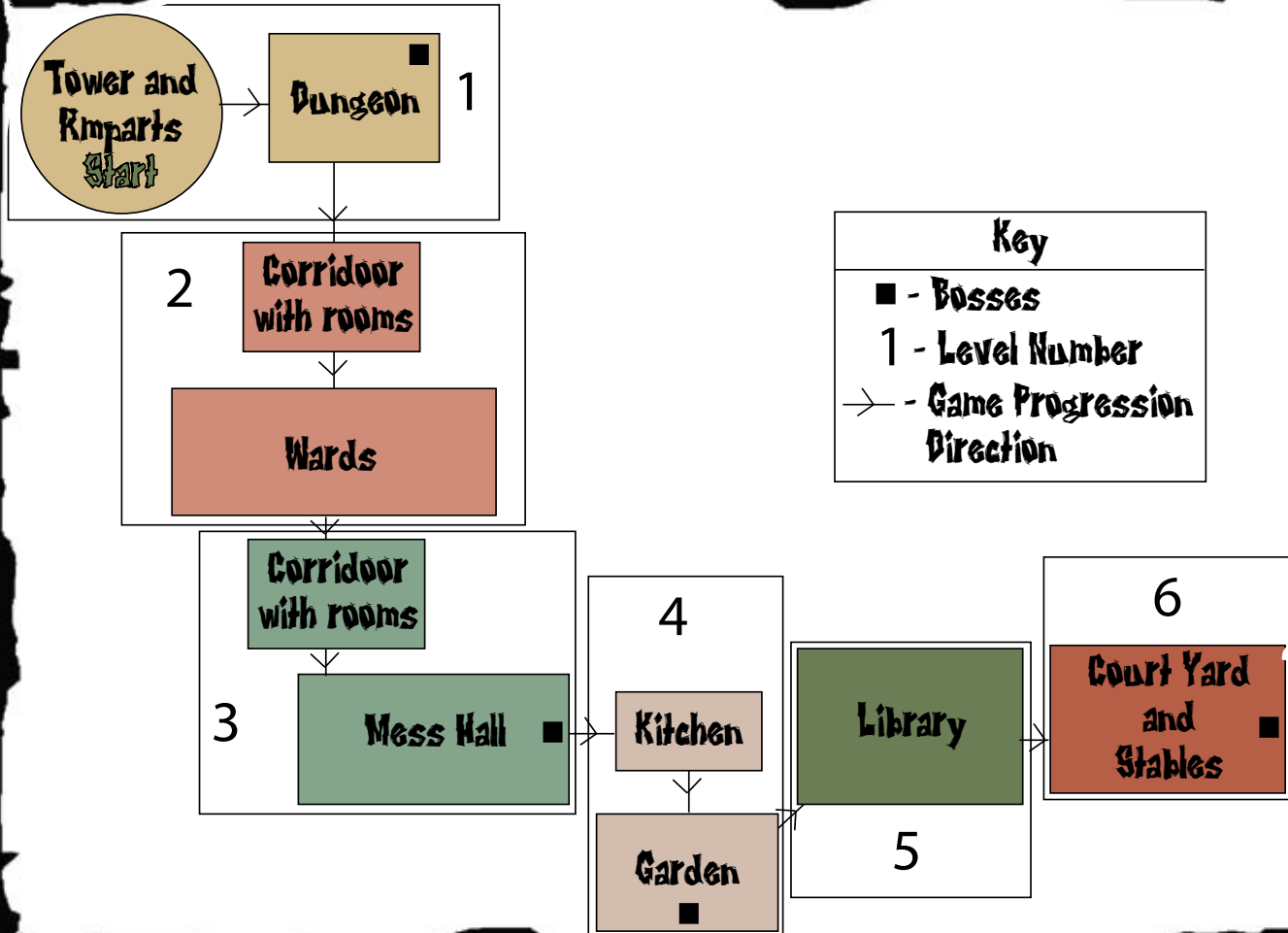


From top left: Dungeon (Level 1.2),
Wards (Level 2), Wards (Level 3)



From top left: Kitchen and Gardens (Level 4),
Library (Level 5), Castle Court Yard (Level 6).

Game Progression



Level 1

Areas: Tower & Ramparts , Guard Room and Dungeon

Boss: Teddy

Type of Weapons: Sniper, Vacuum gun and Flamer

Enemies: Sniper bears and Nurse (Shield) bears

- Guard teddies try and sniper you from the other towers
- Clear out the guard room and dungeon.
- In dungeon you meet boss Teddy

Level 2

Areas: Corridor with generic rooms and Wards

Type of Weapons: Sniper, Vacuum gun and Flamer

Enemies: All previous plus shield and sniper Flippy

Level 3

Areas: Corridor with generic rooms and Mess Hall

Boss: Snoopy

Type of Weapons: Sniper, Vacuum gun, Flamer and Wodstock Cannon

Enemies: All previous plus Flamer Flippy

- Clean out rooms then fight Snoopy

Level 4

Areas: Kitchen and Garden

Boss: Mr Bunnikins

Type of Weapons: Sniper, Vacuum gun, Flamer, Wodstock Cannon and Egg Launcher

Enemies: All previous plus Woodstock Flippy

- Clean out rooms, then fight Mr Bunnikins

Level 5

Areas: Library

Type of Weapons: Sniper, Vacuum gun, Flamer, Wodstock Cannon and Egg Launcher

Enemies: All previous plus Egg launcher teddies

Level 6

Areas: Court Yard and Stables

Boss: The Toymaker

Type of Weapons: Vacuum gun, Flamer, Wodstock Cannon and Egg Launcher

Enemies: theToymaker

- Kill the toymaker and leave H.E.L.L

Characters

The Stories in these character profiles are the basis for the comic strip inteludes that will introduce each character.



Good Luck Bear a.k.a Lucky

Lucky, was once a well-known children's icon, until it was discovered that he was in a relationship with Sunshine Bear. The officials arrested him for statutory rape, however due to the fact that both characters involved were not considered legitimate people it was unclear how and if the rules around the age of consent applied. In the end the case was dismissed, however the bad press associated with the case made it impossible for Lucky to find work. He spiralled down into the depths of the toy underworld of sex, drugs and mob crime. No one knows what finally made him snap it may have been the 20 year anniversary celebration show that he was hired to do, or the very unflattering 'Were Are They Now' documentary. Whatever it was, on the night of the 21st of August 2005 he took an axe into the Kingdom of Caring and proceeded to slaughter the entire Care Bear population. "It was a terrible scene," says one witness, "He just came in and the sky turned dark. It was Love-A-Lot Bear that died first, he came running up to Lucky and tried to hug him. Then Lucky just pulled this axe out. He said 'This is what happens when the world stops caring!' and attacked." Lucky has been confined to the Healing Emporium for Lost Lovelies (H.E.L.L.) for life without the possibility of parole.



The Toymaker

Attack: Spawns generic Melee characters & uses a charge up paint gun based off the bio-rifle

Story: The Toymaker was once known as Barnabas Little. He was a world famous toymaker each of his toys was hand crafted and unique. For decades he made toys forever moulding, carving and painting, then one day his health began to fail. It was then that the idea came to him, to continue his work he needed to live forever like one of his toys.

He became a recluse and for many years nothing was heard of him, until one day the neighbours noticed a terrible odour coming from the premises. The police were called in and they found the mutilated remains of 27 young women and boys. Entering the shop they found row upon row of dolls, they were made of porcelain, wood, led, fabric and every other combination you could think of. It was not until they reached the last room did they find him, a child manikin made of wood and porcelain, sitting amongst 27 other dolls. Many of the dolls were bound and gagged their eyes covered. They arrested him on 27 counts of murder only to find many more bodies beneath the floor; his failed attempts at transferring a human consciousness to a doll. On closer inspection the transferral process was discovered to have serious flaws. Most of the transfers resulted in a degree of insanity and sociopathy, these sociopathic tendencies were most pronounced in those who had been in doll form the longest. In the end most of the Toymakers victims had to be confined to the wards, a few managed to go back with their families only to be committed a short time later. The Toy Maker was charged with 189 counts of murder. He was given an indefinite sentence, to be confined to H.E.L.L. until death, without the possibility of parole.

Mr Bunnikins

Attack: Uses a modified flack cannon that shoots exploding Easter eggs

Story: Mob boss and all round bad guy, managed to get put into H.E.L.L. with the help of the local mob owned psychologist. Usually he lives incognito with a Miss Betty Losenguard, this works to his advantage as it means he almost



always has an alibi. He has been linked to the murder of hundreds of low level criminals and high-level competitors; however only one had enough evidence to be brought before the courts. The court heard that he walked into his home to find his current squeeze, Muffy doing the local milk man. In a fit of rage he pulled out a meat cleaver and hacked them both to death. It was him that managed to import the weapons into H.E.L.L. and then orchestrate the mass break out, however he hadn't counted on the Toymaker using his preparations to bring about a totally different plan.

Snoopy

Attack: Uses a modified Rocket Launcher that shoots Woodstocks

Story: After the murder of his caretaker, Charlie Brown, Snoopy became a recluse obsessed with finding Charlie's killer. He obsessed over every aspect of the crime. Even Woodstock, his faithful companion, couldn't bring snoopy out of his self imposed darkness. It soon became clear that snoopy was a bit unhinged. He had a number of bouts in various psychiatric institutions, but very little could be done for him, particularly since verbal/oral forms of therapy were ineffective. Snoopy finally went over the edge late 2002. It was the anniversary of Charlie's Murder, Linus and Sally had invited Snoopy over for dinner. They hoped that the company would alleviate Snoopy's depression. The orderes had jsut been served and he snapped, they had served up his portion and put the parsley on it just like he liked it. He stared at the parsley and his eyes began to twitch, he pulled the serving fork from the roast and attacked, Linus was found close to the upturned food boll he had been stabbed 47 times in the chest and had a number of defensive wounds. Sally, Snoopy caught near the door, he stabbed her over 100 times then proceeded to devour her corpse. The neighbors heard the screams and called the police. The police arrived in the middle of snoopy devouring Sally. They said he was rabid and uncontrollable, they subdued him and shipped him off to H.E.L.L. he was considered too dangerous to be allowed out to his trial so, he was tried in absentia. On behalf of public safety he was sentenced be an inmate of H.E.L.L. for the rest of his natural life. In true companionship Woodstock chose to accompany Snoopy into his incarceration, forever the faithful companion.



NPCs

Teddies



Purple - Egg launcher teddy, Orange - Flamer teddy, Blue - Nurse (shield) teddy, Yellow - Sniper Teddy

The Boss Teddy uses the Flamer Teddy Base



Happy Tree Friends: Flippy



There are several different classes of flippy, we have the standard flippy that uses the Vacuum gun, a Sniper Flippy, a Flamer Flippy and a Woodstock Flippy.

AI

There are a number of different configurations of AI that we will need to use though the main types are:

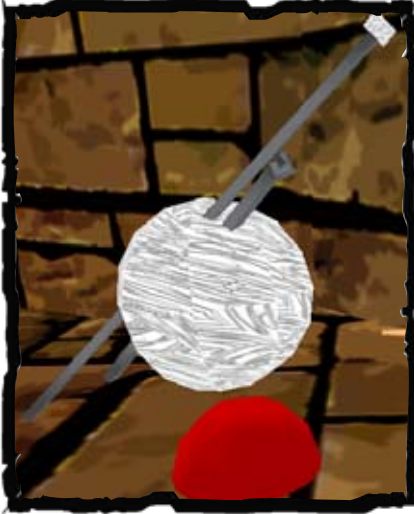
- Proximity and tracking AI so that the Melee characters will follow our protagonist once he has come within their line of sight;
- sniper AI for the guard teddies;
- Attack AI - Weapons attacks;



Game Elements

Items

- Sewing Kits - that act as healing kits, with larger kits for more health and smaller needles and thread for small health top ups.



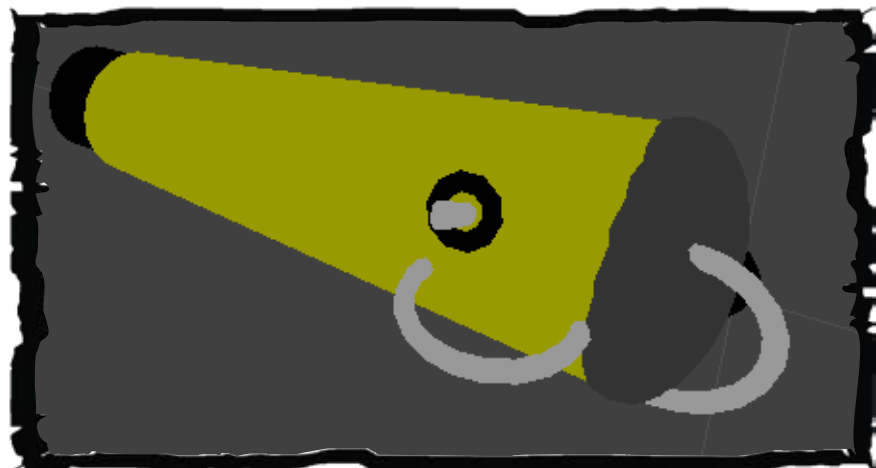
Sniper Ammo



Woodstock Ammo

Weapons and Ammo

- Snoopy customised rocket launcher that shoots Woodstocks
- Mr Bunnikins customised flack cannon that shoots Easter eggs
- The Toymaker customised bio-rifle that shoots paint
- Jack o lanterns - flame thrower and bomb
- Vacuum gun



Woodstock Cannon



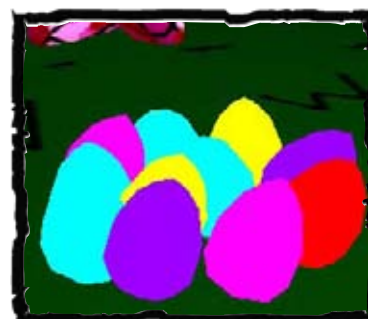
Pumkin Flamer
and
Flamer Ammo



Paint gun



Egg Launcher
and
Egg Ammo



References:

Teddy Scares

<http://www.teddyscares.com>

Serial Killer DB

<http://www.crimezzz.net>

Snoopy Home Page

<http://www.snoopy.com>

Care Bears Wiki

http://en.wikipedia.org/wiki/Care_Bears

Care Bears TV show Wiki

[http://en.wikipedia.org/wiki/The_Care_Bears_\(TV_series\)](http://en.wikipedia.org/wiki/The_Care_Bears_(TV_series))

Care Bears Home Page

<http://www.agkidzone.com/carebears.action>

Vintage Cabbage Patch Kids

<http://www.geocities.com/kakiloge/CabbagePatchRerootOrders.html>

Cabbage Patch Kids Original Clown Costume

http://cgi.ebay.com/RARE-Vintage-Cabbage-Patch-Kid-CLOWN-OUTFIT-CPK-Clothes_0QQitemZ380051549750QQihZ025QQcategoryZ36549QQrdZ1QQssPageNameZWD4VQQcmdZViewItemQQ_trksidZp1638Q2em124#ebayphotohosting

Cabbage Patch Kid baby doll

http://cgi.ebay.com/Cabbage-Patch-Kids-Baby-Doll-Collectible-177_0QQitemZ250277175538QQihZ015QQcategoryZ243QQrdZ1QQssPageNameZWD4VQQcmdZViewItemQQ_trksidZp1638Q2em124%E2%80%9D

Goldilocks Marionette

http://www.toymotion.com/html/goldilocks_marionette.html

YouTube - Happy Tree Friends Eppisode

<http://www.youtube.com/watch?v=RZjTQ87MoEo>

"It Must Be Destroyed" Font

<http://www.dafont.com/it-must-be-destroyed.font>

Unreal Developers Network

<http://udn.epicgames.com/Main/WebHome.html>



Welcome To H.E.L.L.

Art Bible



Contents

Overview & Art Style.....	3
Technical Guide Lines.....	3
Textures.....	4
HUD	5
Screen shots (Version one).....	6
Screen shots (Final)	8
H.E.L.L.....	10
Character Ideas	11
The Library.....	15
Items.....	15
The Toymaker	16
Mr Bunnikins.....	17
Other Characters	18
Story Board.....	19
Comic Strips	20
Visual References	24



Overview & Art Style

Welcome to H.E.L.L. (WTH) is a comic horror; to fit in with this genre and the twisted toy characters are looking at using a cell shaded render. This will add to the comic style of our piece tying it in nicely with the comic interludes.

For consistencies sake the art style we are using for the development of this mod is largely based on children's cartoons and comics, with a definite leaning towards a western rather than Asiatic feel.

In terms of the publications graphic style we are using a combination of the "It Must Be Destroyed" type face for headings (<http://www.dafont.com/it-must-be-destroyed.font>) and Comic Sans for the body text. Both these fonts keep the comic/cartoonish style, and in the case of "It Must Be Destroyed" adding a bit of the sinister.

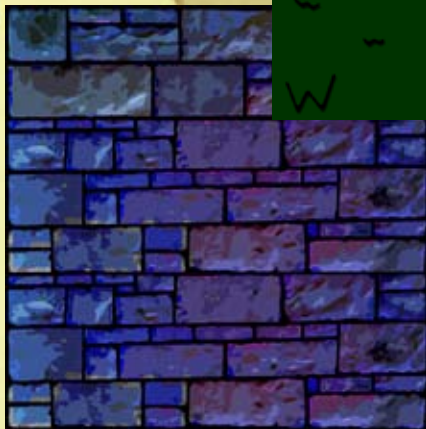
Technical Guide Lines

In the development of this mod we are using a number of programs, for a variety of functions:

- 3D modelling - Maya
- Image and texture manipulation - Adobe Photoshop & Illustrator
- Level design - UT2004 editor
- Comic strip layout - A Comic Life
- Design document layout - Adobe Indesign CS3

Textures

Because of the distinctly comic feel of our mod we customised all our own textures. Using a combination of high contrast, strong black lines and bright colours created very comical textures. Some of the textures were based off real life textures (e.g. the stone walls) others were based off images, earlier textures or created entirely from scratch. We also had both a clean and blood splattered version of many of the textures to add a bit of gore and complexity to the comic image style.

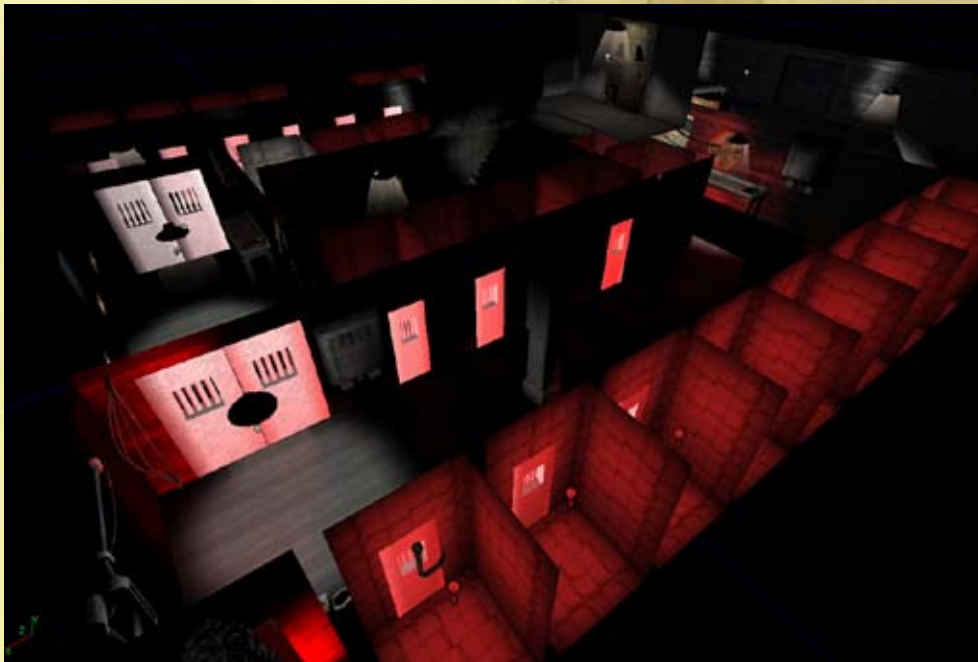


HUD

To keep within the comic style of the game we made a comic HUD, utilising the cate motif. We decided to go with a visual as well as numerical health indicator and custom weapon images.



Screen shots (Version one)

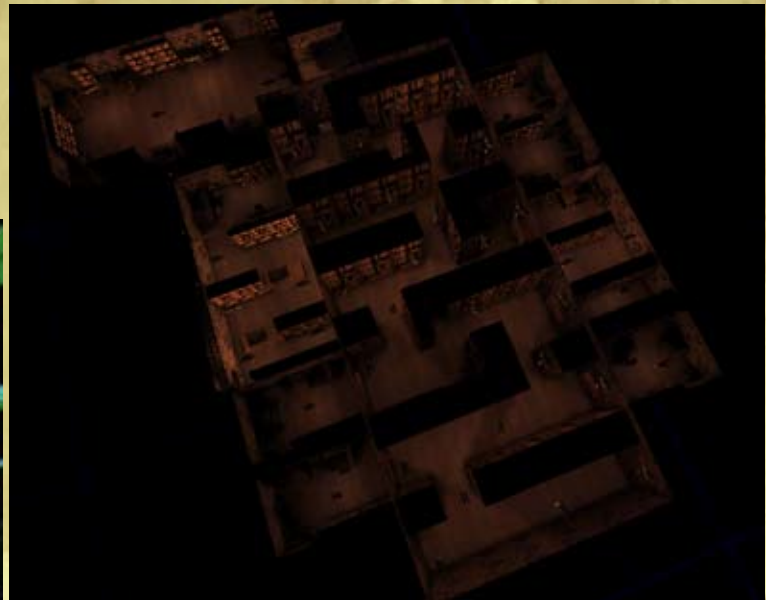


Left: Dungeon
(Level 1.2)
Below Left:
Wards (Level 3)
Below Right:
Wards (Level 2)

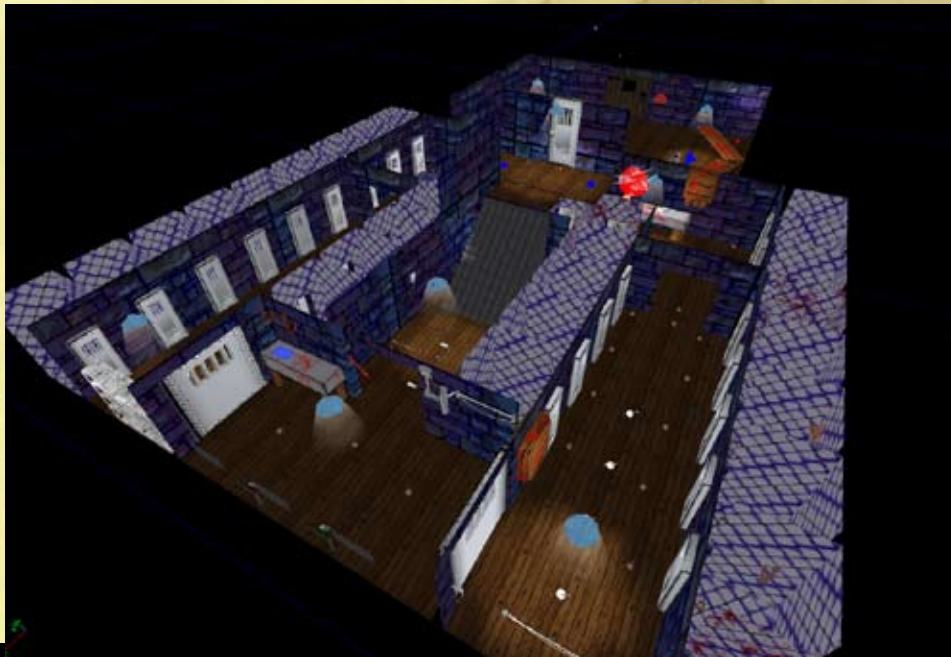




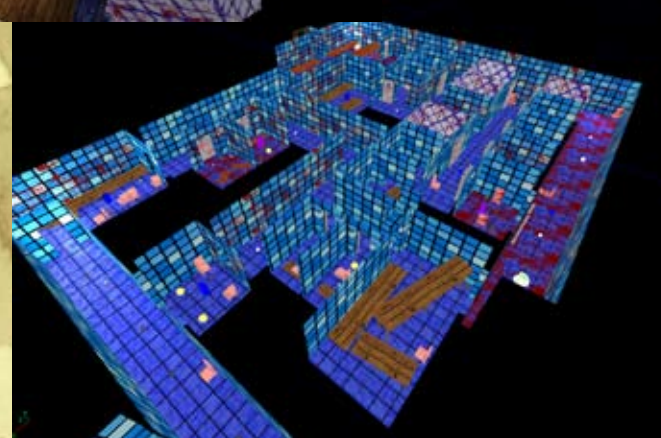
Left & Bottom left:
The Kitchen and Garden
(Level 4)
Below: The Library
(Level 5)



Screen shots (Final)



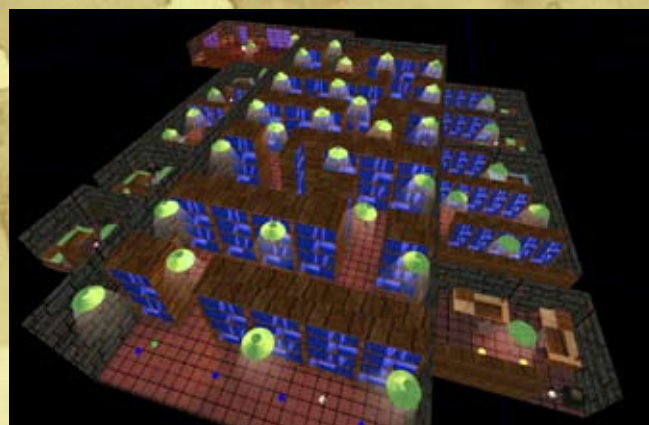
Left: Dungeon
(Level 1.2)
Below Left:
Wards (Level 3)
Below Right:
Wards (Level 2)



Right: H.E.L.L.
Castle (Level 1
& 6).

Below: Kitchen
and garden
(Level 4)

Below right:
Library (Level 5)



HELL.



Character Ideas





His Coming down the
Stairs!

The downy Teddys



THE FUZ!



"FURRY"





Lets Play Doctor



You Up For A Puzil

The Library



Items



Health

The Toymaker

I AM FOREVER A
TOY.

I AM FOREVER
THE TOY MAKER!



Mr Bunnikins



How are you self... Are
you feeling cludy... funk



Dance m's
pretty! Dance!



At last we
knew we could
and this... after
party... none...

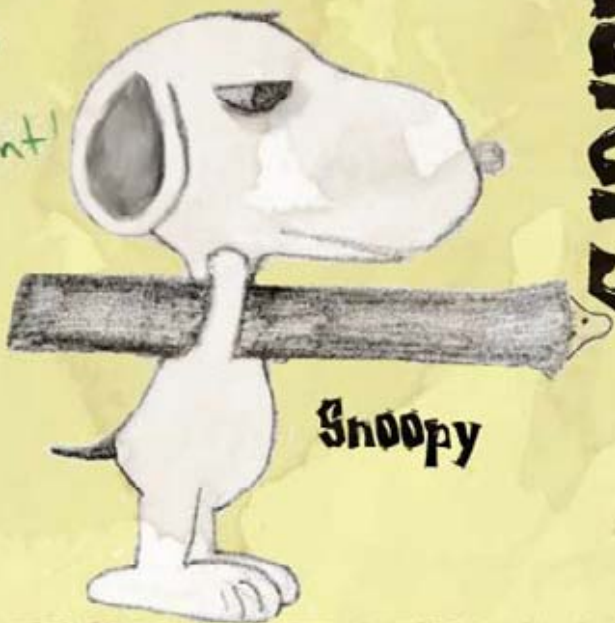


Other Characters



I'm go'n
on a
bear hunt!

Lucky

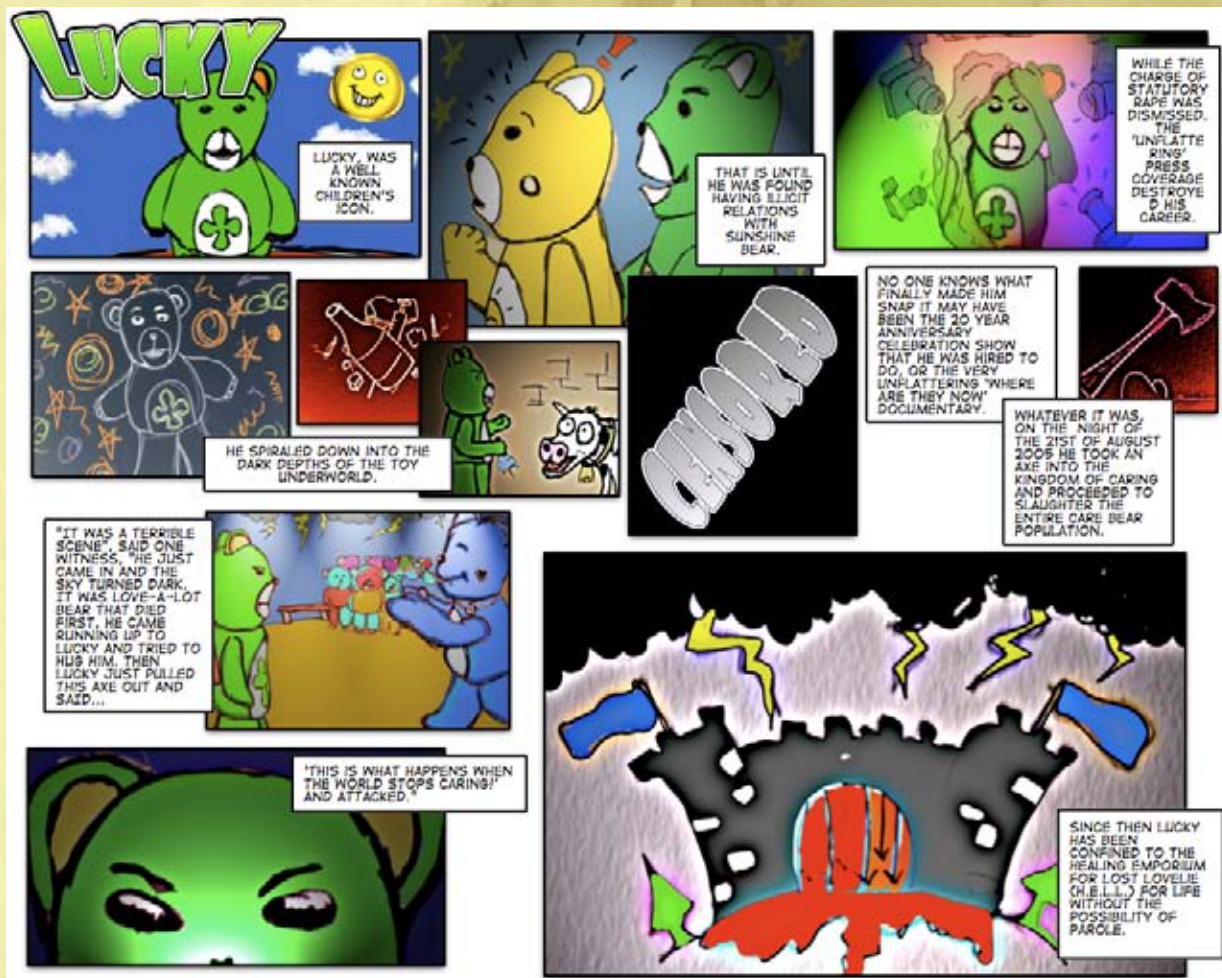


Snoopy

Story Board

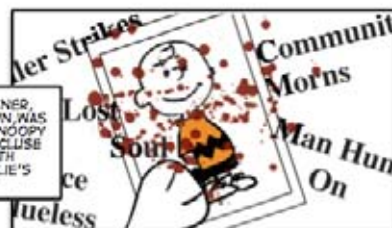


Comic Strips

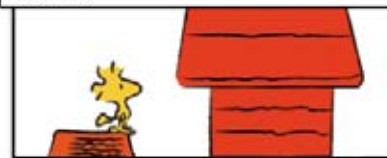




AFTER HIS OWNER, CHARLIE BROWN, WAS MURDERED, SNOOPY BECAME A RECLUSE OBSESSED WITH FINDING CHARLIE'S KILLER.



HE OBSESSED OVER EVERY ASPECT OF THE CRIME. EVEN WOODSTOCK, HIS FAITHFUL COMPANION, COULDN'T BRING SNOOPY OUT OF HIS SELF IMPOSED DARKNESS.



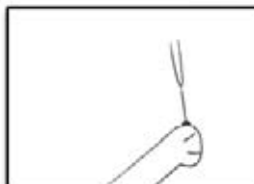
IT SOON BECAME CLEAR THAT SNOOPY WAS A BIT UNHINGED. HE HAD A NUMBER OF BOUTS IN VARIOUS PSYCHIATRIC INSTITUTIONS, BUT VERY LITTLE COULD BE DONE FOR HIM.



SNOOPY FINALLY WENT OVER THE EDGE LATE 2002. IT WAS THE ANNIVERSARY OF CHARLIE'S MURDER. LINUS AND SALLY HAD INVITED SNOOPY OVER FOR DINNER.



THEY HAD SERVED UP HIS PORTION AND PUT THE PARSLEY ON IT JUST LIKE HE LIKED IT. HE STARED AT THE PARSLEY AND HIS EYES BEGAN TO TWITCH.



LINUS WAS FOUND CLOSE TO THE UPTURNED FOOD BOWL HE HAD BEEN STABBED 47 TIMES IN THE CHEST AND HAD A NUMBER OF DEFENSIVE WOUNDS.

HE PULLED THE SERVING FORK FROM THE ROAST AND ATTACKED.



SALLY, SNOOPY CAUGHT NEAR THE DOOR, HE STABBED HER OVER 100 TIMES THEN PROCEEDED TO DEVOUR HER CORPSE.

THE NEIGHBORS HEARD THE SCREAMS AND CALLED THE POLICE. HE WAS CONSIDERED TOO DANGEROUS TO BE ALLOWED OUT TO HIS TRIAL SO, HE WAS TRIED IN ABSENTIA. IN THE INTEREST OF PUBLIC SAFETY HE WAS SENTENCED TO H.E.L.L. FOR THE REST OF HIS NATURAL LIFE. IN TRUE COMPANIONSHIP WOODSTOCK CHOSE TO ACCOMPANY SNOOPY INTO HIS INCARCERATION, FOREVER THE FAITHFUL COMPANION.



MR BUNNIKINS

MR BUNNIKINS IS A MOB BOSS AND ALL ROUND BAD GUY. HE HAS BEEN LINKED TO THE MURDER OF HUNDREDS OF LOW LEVEL CRIMINALS AND HIGH-LEVEL COMPETITORS; HOWEVER ONLY ONE HAD ENOUGH EVIDENCE TO BE BROUGHT BEFORE THE COURTS...



THE COURT HEARD THAT HE WALKED INTO HIS HOME TO FIND HIS CURRENT SQUEEZE, MIFFY DOING THE LOCAL MILK MAN...

THIS HAS ALLOWED HIM TO CONDUCT HIS AFFAIRS WITH LITTLE TROUBLE FROM THE GUARDS, HOWEVER HE HAS GROWN WEARY OF HIS INCARCERATION. IT WAS HIM THAT IMPORTED THE WEAPONS IN TO H.E.L.L. AND THEN ORCHESTRATED THE MASS BREAK OUT, HOWEVER HE HADN'T COUNTED OTHER FACTIONS WITH OTHER PLANS

IN A FIT OF RAGE HE PULLED OUT A MEAT CLEAVER AND HACKED THEM BOTH TO DEATH.

WELCOME TO HELL

WITH THE HELP OF ONE OF THE LOCAL PSYCHOLOGISTS, WHO WAS HAPPY TO MAKE A BIT ON SIDE, HE MANAGED TO GET PUT INTO H.E.L.L. RATHER THAN THE STATE MAXIMUM SECURITY.



The Toymaker



ONCE UPON A TIME THERE LIVED A TOYMAKER, THE GREATEST TOYMAKER THAT LONDON HAD EVER SEEN. THIS TOYMAKER MADE TOYS FOR THE RICH, TOYS FOR THE POOR, TOYS THAT COULD BRIGHTEN EVEN THE DARKEST OF DAYS.



HE WAS NEARING THE END OF A FULL LIFE WHEN HE GAZED AT THE DOLL IN HIS HANDS AND UNDERSTOOD THE DARKNESS THAT WAS COMING. THE CHILDREN WOULD ONCE MORE SUFFER TRAGEDIES, TRIALS AND STRIFE WITHOUT THE SUNLIGHT THAT HIS TOYS COULD BRING. HE VOWED NOT TO LET THIS HAPPEN....



YEARS PAST, THE TOYMAKER'S LITTLE TOYSHOP WAS DARK. THE NAME OF THE TOYMAKER HAD BEEN LOST, ANOTHER CANDLE BLOWN OUT IN A FOREST FIRE.



IT WAS A SQUATTER WHO FOUND THEM, DOLLS, SO MANY DOLLS, AND AT THEIR CENTRE, THE LITTLE TOYMAKER.

HIS SOUL ENTOMBED IN THE BODY OF A TOY, HIS COMPANION, CHILDREN OF LONDON IMPRISONED LIKE HIM.



THE HORROR WROUGHT BY THE LITTLE TOYMAKER HAS BEEN LONG FORGOTTEN BY MOST, THE KEYS TO HIS CELL PASSED FROM ONE JAILER TO THE NEXT, BUT THE DOOR NEVER OPENED.

THAT IS, UNTIL THE QUEST TO EXIST FOREVER STRIKES SOMEONE ELSE'S FANCY, SOMEONE WHO DOESN'T CARE ABOUT THE COST.



Visual References



Copyright © United Feature Syndicate, Inc.
Redistribution in whole or in part prohibited.





Welcome To H.E.L.L. Technical Bible



Contents

WTH Class List	3
Script List	10
References	11



WTH Class List

PACKAGE: WTHgame

Class: WTHgameType

Description: This class specifies the gametype as well as the hud and other descriptive properties such as the default player name.

Extends: OSMGame

Overrides: defaultproperties (HUDType, GameName, DefaultPlayerName)

More Info: The WTHgameType will extend the OSMGame adventure gametype and will specify the custom WTHHUD to use.

Class: WTHMainMenu

Description: This class will create the custom main menu for our mod.

Extends: UT2K4GUIPage

Overrides: InitComponent, ButtonClick, defaultproperties (GUI Objects, Button Labels)

More Info: The InitComponent will invoke the Super.InitComponent to create the menu with the custom GUI Objects. The ButtonClick will control what menu object is clicked.

Class: WTHHUD

Description: This class extends the HUD class and will modify the fonts displayed on the HUD as well as adding a doom style

player portrait as a reference to the players physical status.

Extends: HudBDeathMatch

Overrides: DrawHud, defaultproperties (FontArrayNames)

More Info: The correct font will be set in the default properties and DrawHud will be modified to display the doom style player portrait. For example at 75-100 health a healthy portrait will be displayed, at 50-75 health a slightly damaged portrait will be displayed, etc.

Class: WTHhealthYarn

Description: This class will give health to the player and look like a ball of wool with sewing needles going through it.

Extends: Pickup

Overrides: Touch, defaultproperties (Mesh)

More Info: The WTHhealthYarn will be identical to a normal health pack with a different mesh.

Class: WTHhealthYarnMini

Description: This class will give health to the player and look like a ball of wool with sewing needles going through it.

Extends: Pickup

Overrides: Touch, defaultproperties (Mesh)

More Info: The WTHhealthYarnMini will be identical to a normal health pack with a different mesh.

Class: WTHhud

Description: This class extends the HUD

class and will modify the fonts displayed on the HUD as well as adding a doom style player portrait as a reference to the players physical status.

Extends: HUD

Overrides: DrawHud, defaultproperties (FontArrayNames)

More Info: The correct font will be set in the default properties and DrawHud will be modified to display the doom style player portrait. For example at 75-100 health a healthy portrait will be displayed, at 50-75 health a slightly damaged portrait will be displayed, etc.

NEW Class: WTHmutator

Description: This class will replace many weapons of the game with our custom weapons as well as the ammo.

Extends: Mutator

Overrides: CheckReplacement

More Info: Specific weapons will be changed to their WTH equivalent, including the inventory class and the pickup class. The ammo will also be changed. This mutator can be used for other games too.

NEW Class: WTHplayer

Description: This is the default player class.

Extends: xPawn

Overrides: defaultproperties(Health, RequiredEquipment)

More Info: The default player will have the vacuum cleaner as the starting weapon and will have the appearance of Lucky.

PACKAGE: WTHcharactersClass

NEW Class: WTHbaseCharacter

Description: This is the base character class that all other enemies will be extended from

Extends: ScriptedCharacter

Overrides: defaultproperties(Orders, intelligence, attitudetoplayer, GoodieDrop)

More Info: This sets the enemies to a basic dumb intelligence type and gives them all a vacuum cleaner by default.

NEW Class: WTHteddy

Description: This is the base teddy class that other teddy's will extend.

Extends: WTHbaseCharacter

Overrides: defaultproperties(OverrideHealth, Mesh, Skins)

More Info: This class changes the mesh of the enemy to a teddy and gives it a small amount of health.

NEW Class: WTHteddyEgg

Description: This class extends the base teddy and gives him an egg launcher and a different skin.

Extends: WTHteddy

Overrides: defaultproperties(GoodieDrop, Skins)

More Info: This class changes the weapon of the teddy to an egg launcher and gives it a different skin.

NEW Class: WTHteddyFlamer

Description: This class extends the base teddy and gives him a pumpkin flamer and a

different skin.

Extends: WTHteddy

Overrides: defaultproperties(GoodieDrop, Skins)

More Info: This class changes the weapon of the teddy to a Pumpkin Flamer and gives it a different skin.

NEW Class: WTHteddySniper

Description: This class extends the base teddy and gives him a sniper rifle and a different skin.

Extends: WTHteddy

Overrides: defaultproperties(GoodieDrop, Skins)

More Info: This class changes the weapon of the teddy to a sniper rifle and gives it a different skin.

NEW Class: WTHbossTeddy

Description: This class extends the base teddy and gives him a sniper rifle and a different skin.

Extends: WTHteddy

Overrides: defaultproperties(Override Health, AttitudeToPlayer, Intelligence, bIsBoss)

More Info: This class makes the base teddy more intelligent and also sets the boss flag.

NEW Class: WTHflippy

Description: This is the base flippy class that other flippy's will extend.

Extends: WTHbaseCharacter

Overrides: defaultproperties(OverrideHealth, Mesh, Skins)

More Info: This class changes the mesh of the enemy to a flippy and gives it a small amount of health.

NEW Class: WTHflippyFlamer

Description: This class extends the base flippy and gives him a pumpkin flamer.

Extends: WTHflippy

Overrides: defaultproperties(GoodieDrop)

More Info: Changes the used weapon and dropped weapon to the pumpkin flamer.

NEW Class: WTHflippySniper

Description: This class extends the base flippy and gives him a sniper rifle.

Extends: WTHflippy

Overrides: defaultproperties(GoodieDrop)

More Info: Changes the used weapon and dropped weapon to the Woodstock launcher.

NEW Class: WTHflippyWoodstock

Description: This class extends the base flippy and gives him a Woodstock launcher.

Extends: WTHflippy

Overrides: defaultproperties(GoodieDrop)

More Info: Changes the used weapon and dropped weapon to the Woodstock launcher.

NEW Class: WTHbunnikins

Description: This is the class for the bunnikins boss.

Extends: WTHbaseCharacter

Overrides: defaultproperties(OverrideHealth, AttitudeToPlayer, Intelligence, bIsBoss, Mesh)

More Info: This class will create the bunnikins boss and give him a human-like intelligence, it will also set his mesh, skin and health to high value, and also give him an egg gun.

NEW Class: WTHsnoopy

Description: This is the class for the snoopy boss.

Extends: WTHbaseCharacter

Overrides: defaultproperties(Override Health, AttitudeToPlayer, Intelligence, bIsBoss, Mesh)

More Info: This class will create the snoopy boss and give him a human-like intelligence, it will also set his mesh, skin and health to high value, and also give him a Woodstock launcher.

NEW Class: WTHtoyMaker

Description: This is the class for the toy maker boss.

Extends: WTHbaseCharacter

Overrides: defaultproperties(Override Health, AttitudeToPlayer, Intelligence, bIsBoss, Mesh)

More Info: This class will create the toy maker boss and give him a human-like intelligence, it will also set his mesh, skin and health to high value, and also give him a paint gun.

PACKAGE: WTHweaponClass

Class: WTHsniperRiflePickup

Description: This class extends the

UTWeaponPickup class. The pickup will look like a custom model of a sniper rifle and when it is collected, the WTHsniperRifle will be added to the inventory.

Extends: UTWeaponPickup

Overrides: defaultproperties (DrawType, StaticMesh, InventoryType)

More Info: The DrawType will be set to DT_StaticMesh so that a static mesh can be used as the visible model of the weapon. The static mesh will then be set to the third person view of the WTHsniperRifle mesh. When the pickup is collected, the inventory type WTHsniperRifle will be added to the inventory and the custom PickupMessage will be seen.

Class: WTHsniperRifle

Description: This class extends the original SniperRifle. The launcher will look like a original sniper rifle and it fires the standard SniperFire class.

Extends: ClassisSniperRifle

Overrides: defaultproperties (Mesh, ItemName)

More Info: The mesh will be changed to the first person view of the WTHsniperRifle and the item name will be set to "Sniper Rifle". The

Class: WTHeggLauncherPickup

Description: This class extends the UTWeaponPickup class. The pickup will look like a custom model and when it is collected, the WTHeggLauncher will be added to the inventory.

Extends: UTWeaponPickup

Overrides: defaultproperties (DrawType, StaticMesh, InventoryType)

More Info: The DrawType will be set to DT_StaticMesh so that a static mesh can be used as the visible model of the weapon. The static mesh will then be set to the third person view of the WTHeggLauncher mesh. When the pickup is collected, the inventory type WTHeggLauncher will be added to the inventory and the custom PickupMessage will be seen.

Class: WTHeggLauncher

Description: This class extends the ONSGrenadeLauncher. The launcher will look like a custom weapon and it fires the WTHeggLauncherFire class.

Extends: ONSGrenadeLauncher

Overrides: defaultproperties (Mesh, ItemName, FireModeClass, AttachmentClass)

More Info: The mesh will be changed to the first person view of the WTHeggLauncher and the item name will be set to "Easter Egg Launcher". The FireModeClass will be set to WTHeggLauncherFire and AttachmentClass will be set to WTHeggLauncherAttachment as well.

Class: WTHeggLauncherAttachment

Description: This class extends the ONSGrenadeAttachment to allow the easter egg launcher to have the same attachment properties as a grenade launcher, such as third person effects and

animations.

Extends: ONSGrenadeAttachment

Overrides: defaultproperties (Mesh)

More Info: The class will be identical to the ONSGrenadeAttachment with the exception of the mesh for the third person view of the egg launcher.

Class: WTHeggLauncherFire

Description: This is the fire class that will be activated when the WTHeggLauncher is fired.

Extends: ProjectileFire

Overrides: InitEffects, defaultproperties (FireRate, ProjectileClass, FlashEmitterClass)

More Info: InitEffects attaches the muzzle flash FlashEmitterClass to the specified socket. The ProjectileClass sets the WTHeggLauncherFire that is fired from the weapon.

Class: WTHeggLauncherProjectile

Description: This class extends the Grenade and changes the mesh of the grenade to an easter egg as well as changing the explosion created by the grenade.

Extends: Grenade

Overrides: PostNetBeginPlay, Destroyed, defaultproperties (Mesh, Color)

More Info: PostNetBeginPlay and Destroyed are to handle the new custom trail left by the projectile.

Class: WTHvacuumGunPickup

Description: This class extends the

UTWeaponPickup class. The pickup will look like a futuristic vacuum cleaner and when it is collected, the WTHvacuumGun will be added to the inventory.

Extends: UTWeaponPickup

Overrides: defaultproperties (DrawType, StaticMesh, InventoryType)

More Info: The DrawType will be set to DT_StaticMesh so that a static mesh can be used as the visible model of the weapon. The static mesh will then be set to the third person view of the WTHvacuumGun mesh. When the pickup is collected, the inventory type WTHvacuumGun will be added to the inventory and the custom PickupMessage will be seen.

Class: WTHvacuumGun

Description: This class extends the ShieldGun. The launcher will look like a vacuum cleaner and it fires the WTHvacuumGunFire class.

Extends: ShieldGun

Overrides: defaultproperties (Mesh, ItemName, FireModeClass, AttachmentClass)

More Info: The mesh will be changed to the first person view of the WTHvacuumGun and the item name will be set to "Vacuum Cleaner". The FireModeClass will be set to WTHvacuumGunFire and AttachmentClass will be set to WTHvacuumGunAttachment as well.

Class: WTHvacuumGunAttachment

Description: This class extends the

ShieldAttachment to allow the vacuum cleaner to have the same attachment properties as a shield gun, such as third person effects and animations.

Extends: ShieldAttachment

Overrides: defaultproperties (Mesh)

More Info: The class will be identical to the BioAttachment with the exception of the mesh for the third person view of the paint gun.

Class: WTHvacuumGunFire

Description: This is the fire class that will be activated when the WTHvacuumGun is fired.

Extends: ShieldFire

Overrides: InitEffects, defaultproperties (FireRate, FlashEmitterClass)

More Info: InitEffects attaches the muzzle flash FlashEmitterClass to the specified socket. The WTHvacuumGun has no projectile.

Class: WTHwoodstockLauncherPickup

Description: This class extends the UTWeaponPickup class. The pickup will look like a simple cannon and when it is collected, the WTHwoodstockLauncher will be added to the inventory.

Extends: UTWeaponPickup

Overrides: defaultproperties (DrawType, StaticMesh, InventoryType)

More Info: The drawtype will be set to DT_StaticMesh so that a static mesh can be used as the visible model of the weapon. The static mesh will then be set to the third person view of the

WTHwoodstockLauncher mesh. When the pickup is collected, the inventory type WTHwoodstockLauncher will be added to the inventory and the custom PickupMessage will be seen.

Class: WTHwoodstockLauncher

Description: This class extends the rocket launcher. The launcher will look like a simple cannon and it fires the WTHwoodstockLauncherFire class.

Extends: RocketLauncher

Overrides: defaultproperties (Mesh, ItemName, FireModeClass, AttachmentClass)

More Info: The mesh will be changed to the first person view of the WTHwoodstockLauncher and the item name will be set to "Woodstock Launcher". The FireModeClass will be set to WTHwoodstockFire and AttachmentClass will be set to WTHwoodstockLauncherAttachment as well.

Class:

WTHwoodstockLauncherAttachment

Description: This class extends the RocketAttachment to allow the Woodstock launcher to have the same attachment properties as a rocket launcher, such as third person effects and animations.

Extends: RocketAttachment

Overrides: defaultproperties (Mesh)

More Info: The class will be identical to the RocketAttachment with the exception of the mesh for the third person view of

the launcher.

Class: WTHwoodstockFire

Description: This is the fire class that will be activated when the WTHwoodstockLauncher is fired.

Extends: ProjectileFire

Overrides: InitEffects, defaultproperties (FireRate, ProjectileClass, FlashEmitterClass)

More Info: InitEffects attaches the muzzle flash FlashEmitterClass to the specified socket. The ProjectileClass sets the WTHwoodstockProjectile that is fired from the weapon.

Class: WTHwoodstockProjectile

Description: This class extends the RocketProjectile and modifies the emitter trail and the mesh of the projectile launched to a model of woodstock.

Extends: RocketProjectile

Overrides: PostNetBeginPlay, Destroyed, defaultproperties (Mesh)

More Info: PostNetBeginPlay and Destroyed are to handle the custom trail left by the new projectile. The mesh of the projectile is set to the Woodstock mesh.

Class: WTHpaintGunPickup

Description: This class extends the UTWeaponPickup class. The pickup will look like a paint gun and when it is collected, the WTHpaintGun will be added to the inventory.

Extends: UTWeaponPickup

Overrides: defaultproperties (DrawType, StaticMesh, InventoryType)

More Info: The drawtype will be set to DT_StaticMesh so that a static mesh can be used as the visible model of the weapon. The static mesh will then be set to the third person view of the WTHpaintGun mesh. When the pickup is collected, the inventory type WTHpaintGun will be added to the inventory and the custom PickupMessage will be seen.

Class: WTHpaintGun

Description: This class extends the biorifle. The launcher will look like a paint gun and it fires the WTHpaintGunFire class.

Extends: BioRifle

Overrides: defaultproperties (Mesh, ItemName, FireModeClass, AttachmentClass)

More Info: The mesh will be changed to the first person view of the WTHpaintGun and the item name will be set to "Paint Gun". The FireModeClass will be set to WTHpaintGunFire and AttachmentClass will be set to WTHpaintGunAttachment as well.

Class: WTHpaintGunAttachment

Description: This class extends the BioAttachment to allow the paint gun to have the same attachment properties as a bio rifle, such as third person effects and animations.

Extends: BioAttachment

Overrides: defaultproperties (Mesh)

More Info: The class will be identical to the BioAttachment with the exception of the mesh for the third person view of the paint gun.

Class: WTHpaintGunFire

Description: This is the fire class that will be activated when the WTHpaintGun is fired.

Extends: ProjectileFire

Overrides: InitEffects, defaultproperties (FireRate, ProjectileClass, FlashEmitterClass)

More Info: InitEffects attaches the muzzle flash FlashEmitterClass to the specified socket. The ProjectileClass sets the WTHpaintGunProjectile that is fired from the weapon.

Class: WTHpaintGunProjectile

Description: This class extends the BioGlob and modifies the emitter trail and the color of the projectile to be a random color generated by a function within the class.

Extends: BioGlob

Overrides: PostNetBeginPlay, Destroyed, defaultproperties (Color)

More Info: A new function will be added to the class to generate a random color for each projectile fired. PostNetBeginPlay and Destroyed are to handle the custom trail left by the new projectile.

NEW Class: WTHpumpkinFlamerPickup

Description: This class extends the UTWeaponPickup class. The pickup will

look like a pumpkin flamer and when it is collected, the WTHpumpkinFlamer will be added to the inventory.

Extends: UTWeaponPickup

Overrides: defaultproperties (DrawType, StaticMesh, InventoryType)

More Info: The drawtype will be set to DT_StaticMesh so that a static mesh can be used as the visible model of the weapon. The static mesh will then be set to the third person view of the WTHpumpkinFlamer mesh. When the pickup is collected, the inventory type WTHpumpkinFlamer will be added to the inventory and the custom PickupMessage will be seen.

NEW Class: WTHpumpkinFlamer

Description: This class extends the rocket launcher. The launcher will look like a pumpkin flamer and it fires the WTHpumpkinFlamerFire class.

Extends: RocketLauncher

Overrides: defaultproperties (Mesh, ItemName, FireModeClass, AttachmentClass)

More Info: The mesh will be changed to the first person view of the WTHpumpkinFlamer and the item name will be set to "Pumpkin Flamer". The FireModeClass will be set to WTHpumpkinFlamerFire and AttachmentClass will be set to WTHpumpkinFlamerAttachment as well.

NEW Class:

WTHpumpkinFlamerAttachment

Description: This class extends the RocketAttachment to allow the pumpkin flamer to have the same attachment properties as a rocket launcher, such as third person effects and animations.

Extends: RocketAttachment

Overrides: defaultproperties (Mesh)

More Info: The class will be identical to the RocketAttachment with the exception of the mesh for the third person view of the pumpkin flamer.

NEW Class: WTHpumpkinFlamerFire

Description: This is the fire class that will be activated when the WTHpumpkinFlamer is fired.

Extends: ProjectileFire

Overrides: InitEffects, defaultproperties (FireRate, ProjectileClass, FlashEmitterClass)

More Info: InitEffects attaches the muzzle flash FlashEmitterClass to the specified socket. The ProjectileClass sets the WTHpumpkinFlamerProjectile that is fired from the weapon.

NEW Class:

WTHpumpkinFlamerProjectile

Description: This class extends the Projectile and is a complete custom projectile.

Extends: Projectile

Overrides: PostNetBeginPlay, Destroyed, defaultproperties

More Info: This is a new projectile that shoots out flame balls at a short distance but for good damage.

Script List

WTHgameType.uc: This sets up the game to be a single player adventure.

WTHMainMenu.uc: This creates the custom game menu.

WTHHUD.uc: This makes our hud with our player portrait.

WTHhealthYarn.uc: This will give health to the player when touched.

WTHhud.uc: This will add a doom style player portrait as a reference to the players physical status.

WTHsniperRiflePickup.uc: This pickup will look like a custom model of a sniper rifle.

WTHsniperRifle.uc: The rifle will look like a custom sniper rifle.

WTHsniperRifleAttachment.uc: This will add the third person view of the sniper rifle.

WTHeggLauncherPickup.uc: This pickup will look like a custom model of our egg launcher.

WTHeggLauncher.uc: The rifle will look like our egg launcher,

WTHeggLauncherAttachment.uc: This will add the third person view of the egg launcher.

WTHeggLauncherFire.uc: This will cause eggs to come out when fired.

WTHeggLauncherProjectile.uc: This is the egg projectile that will be launched.

WTHvacuumGunPickup.uc: This pickup will look like a custom model of our vacuum gun.

WTHvacuumGun.uc: The vacuum gun will act similarly to a shield.

WTHvacuumGunAttachment.uc: This will add the third person view of the vacuum gun.

WTHvacuumGunFire.uc: This will suck with the vacuum gun.

WTHwoodstockLauncherPickup.uc: This pickup will look like a custom model of our Woodstock launcher.

WTHwoodstockLauncher.uc: This will launch our Woodstock projectile like a rocket.

WTHwoodstockLauncherAttachment.uc: This will add the third person view of the launcher.

WTHwoodstockFire.uc: This will launch the Woodstock.

WTHwoodstockProjectile.uc: This is the rocket-like Woodstock projectile.

WTHpaintGunPickup.uc: This pickup will look like a custom model of our paint gun.

WTHpaintGun.uc: This paintgun will act like a biorifle with multicoloured blobs.

WTHpaintGunAttachment.uc: This will add the third person view of the paint gun.

WTHpaintGunFire.uc: This will fire the paint from the gun.

WTHpaintGunProjectile.uc: This is the paint projectile that will change color with each fire.

ADDED

WTHmutator.uc: This will change standard weapons to WTH weapons.

WTHplayer.uc: This will set the player.

WTHbaseCharacter.uc: This is the base character for enemies.

WTHteddy.uc: This is the base teddy character.

WTHteddyEgg.uc: This is the teddy with the egg gun.

WTHteddyFlamer.uc: This is the teddy with the flamer.

WTHteddySniper.uc: This is the teddy with the sniper rifle.

WTHbossTeddy.uc: This is the boss teddy with the flamer.

WTHflippy.uc: This is the base flippy

character.

WTHflippyFlamer.uc: This is flippy with a flamer.

WTHflippySniper.uc: This is flippy with a sniper rifle.

WTHflippyWoodstock.uc: This is flippy with a Woodstock launcher.

WTHbunnikins.uc: This is the boss bunnikins.

WTHsnoopy.uc: This is the boss Snoopy.

WTHtoyMaker.uc: This is the boss toy maker.

WTHpumpkinFlamerPickup.uc: This is the pickup for the pumpkin flamer weapon.

WTHpumpkinFlamer.uc: This is the pumpkin flamer inventory weapon.

WTHpumpkinFlamerAttachment.uc: This is the attachment for 3rd person views of the flamer.

WTHpumpkinFlamerFire.uc: This is the fire class of the flamer to control the firing and projectile.

WTHpumpkinFlamerProjectile.uc: This is the flame projectile.

References

3DBuzz VTM's and website, "MyMod" Example
mod, UDN.



WELCOME TO H.E.L.L.

FINAL ADDENDUM



Contents

Final Addendum	3
Weapons and Attacks	3
Characters	3
Power ups.....	3

Final Addendum

This is the final addendum to the design document containing the changes we made at this point and the justification for those changes. Most of the changes were due to time restraints and the priority to make the mod work smoothly over more custom content.

Game Plan/Progression

We changed the guard house into the castle ramparts, as it was clear that the ramparts could play a similar role in the story as well as being a little more interesting, typographically speaking.

Weapons and Attacks

Due to the limitations of the UT animation for melee attack, we decided to cut out the melee and move to using the Vacuum (shield) gun as our melee like equivalent, subsequently many of our weapons, characters and generic bots had to be altered from the original design to accommodate this.

In the end we had to consider time restraints and the desire to get all the weapons functioning properly and decided to focus on six main weapons rather than the plethora we initially envisioned.

Characters

We felt we did not have time to fully develop and implement all six bosses so we decided to remove the three weakest ones, these were:

- Lintle
- Bob
- Goldilocks

These characters didn't have the humour, character development or interest that the others held so we felt that they were expendable.

In terms of generic characters, we had to recognise again time limitations and focused on the implementation of two base generic characters (Teddy & Flippy) with a range of classes to each. This means that many of the smaller characters and little bots had to be dropped. This altered which types of AI we utilised.

Power ups

In the end due to time limitations we didn't have time to implement the range of power-ups we would have liked to use. The amount of time required to investigate the way the pill could work and the possible effects could not have fit in with the schedule of the third assignment. Instead we focussed on getting our other pickups to function smoothly.

Classes and Scripts

- Class: WTHtripsyPill
- Class: WTHsniperRifleAttachment
- Class: WTHknifePickup
- Class: WTHknife
- Class: WTHknifeAttachment
- Class: WTHknifeFire
- Script: WTHtripsyPill
- Script: WTHsniperRifleAttachment
- Script: WTHknifePickup
- Script: WTHknife
- Script: WTHknifeAttachment
- Script: WTHknifeFire

The tripsy pill was removed from the game as mentioned earlier due to time constraints. The knife was removed because creating a melee weapon with custom animations and complete custom weapon code was definitely beyond our time frame.