

wCIS 488: Game Design II – Winter 2008  
Instructor: Bruce Maxim, Ph.D.

## **Assignment II**

# **Space Exterminators**

Game Design and Specifications

Nabil AlRamli  
Brian Rodriguez  
Jim Nardichia  
Said Ketchman

# Table of Contents

1	Design History .....	6
2	Game Overview .....	7
2.1	Game Concept .....	7
2.2	Feature Set .....	7
2.3	Genre .....	7
2.4	Target Audience .....	7
2.5	Game Flow Summary .....	8
2.6	Look and Feel.....	8
2.7	Project Scope.....	10
2.7.1	Number of locations.....	10
2.7.2	Number of levels .....	10
2.7.3	Number of NPC's .....	10
2.7.4	Number of weapons.....	11
2.7.4.1	Tranquilizer Rifle.....	11
2.7.4.2	Sticky Gun.....	11
2.7.4.3	Engine Gun .....	12
3	Game Play and Mechanics .....	13
3.1	Gameplay.....	13
3.1.1	Game Progression .....	13
3.1.2	Mission/challenge Structure .....	13
3.1.3	Objectives .....	13
3.1.4	Play Flow.....	13
3.2	Mechanics .....	14
3.3	Physics .....	14
3.3.1	Movement .....	14
3.3.1.1	General Movement .....	14
3.3.1.2	Other Movement.....	14
3.3.2	Objects.....	14
3.3.2.1	Picking Up Objects.....	14
3.3.3	Actions.....	14
3.3.3.1	Switches and Buttons .....	14
3.3.4	Combat .....	14
3.4	Screen Flow .....	16
3.4.1	Screen Flow Chart.....	16
3.4.2	Screen Descriptions.....	16
3.4.2.1	Loading Screen .....	16
3.4.2.2	Main Menu Screen .....	17
3.4.2.3	Host Screen.....	19
3.5	Game Options.....	20
3.6	Replaying and Saving.....	20
3.7	Cheats and Easter Eggs.....	20

4	Story, Setting and Character .....	21
4.1	Story and Narrative .....	21
4.1.1	Back story .....	21
4.1.2	Plot Elements.....	21
4.1.3	Game Progression .....	21
4.1.4	License Considerations.....	21
4.1.5	Cut Scenes .....	21
4.1.5.1	Cut scene #1 .....	21
4.1.5.1.1	Actors .....	22
4.1.5.1.2	Description .....	22
4.1.5.1.3	Storyboard.....	22
4.1.5.1.4	Script .....	22
4.1.5.2	Cut scene #2 .....	22
4.1.5.2.1	Actors .....	22
4.1.5.2.2	Description .....	22
4.1.5.2.3	Storyboard.....	22
4.1.5.2.4	Script .....	22
4.2	Game World .....	22
4.2.1	General look and feel of world.....	22
4.2.2	Area #1 .....	23
4.2.2.1	General Description.....	23
4.2.2.2	Physical Characteristics .....	23
4.2.2.3	Levels that use area.....	23
4.2.2.4	Connections to other areas .....	23
4.2.3	Area #2 .....	23
4.2.3.1	General Description.....	23
4.2.3.2	Physical Characteristics .....	23
4.2.3.3	Levels that use area.....	23
4.2.3.4	Connections to other areas .....	23
4.3	Characters .....	24
4.3.1	Character 1: Earl.....	24
4.3.1.1	Back story .....	24
4.3.1.2	Personality.....	24
4.3.1.3	Look .....	24
4.3.1.3.1	Physical characteristics.....	24
4.3.1.3.2	Animations .....	25
4.3.1.4	Special Abilities.....	26
4.3.1.5	Relevance to game story .....	28
4.3.1.6	Relationship to other characters.....	28
4.3.1.7	Statistics .....	28
4.3.2	Character 2: Bugg .....	29
4.3.2.1	Back story .....	29
4.3.2.2	Personality.....	29
4.3.2.3	Look .....	29
4.3.2.3.1	Physical characteristics.....	29
4.3.2.3.2	Animations .....	30
4.3.2.4	Special Abilities.....	30
4.3.2.5	Relevance to game story .....	30

4.3.2.6	Relationship to other characters.....	30
4.3.2.7	Statistics .....	30
5	Levels.....	32
5.1	Level #1.....	32
5.1.1	Synopsis.....	32
5.1.2	Introductory Material.....	32
5.1.3	Objectives.....	33
5.1.4	Physical Description.....	33
5.1.5	Map .....	33
5.1.6	Critical Path .....	34
5.1.7	Encounters.....	34
5.1.8	Level Walkthrough .....	34
5.1.9	Closing Material.....	35
5.2	Level #2.....	35
5.2.1	Synopsis.....	35
5.2.2	Introductory Material.....	35
5.2.3	Objectives.....	35
5.2.4	Physical Description.....	35
5.2.5	Map .....	35
5.2.6	Critical Path .....	35
5.2.7	Encounters.....	35
5.2.8	Level Walkthrough .....	36
5.2.9	Closing Material.....	36
6	Interface .....	37
6.1	Visual System.....	37
6.1.1	HUD - What controls .....	37
6.1.2	Menus.....	38
6.1.3	Rendering System.....	39
6.1.4	Camera .....	39
6.1.5	Lighting Models .....	39
6.2	Control System .....	40
6.3	Audio .....	40
6.4	Music.....	40
6.5	Sound Effects.....	40
6.6	Help System.....	40
7	Artificial Intelligence.....	41
7.1	Opponent AI .....	41
7.2	Enemy AI – Villains and Monsters .....	41
7.2.1	Player and Collision Detection .....	41
7.2.2	Pathfinding .....	41
8	Technical.....	42
8.1	Target Hardware .....	42
8.2	Development hardware and software .....	42
8.3	Development procedures and standards.....	43
8.4	Game Engine .....	43
8.5	Network.....	43
8.6	Scripting Language .....	43
9	Game Art .....	44

9.1	Concept Art .....	44
9.2	Style Guides.....	44
9.3	Characters .....	44
9.4	Environments .....	44
9.5	Equipment .....	44
9.6	Cut scenes.....	44
9.7	Miscellaneous.....	44
10	Secondary Software .....	44
10.1	Editor .....	44
10.2	Installer.....	44
10.3	Update software.....	44
11	Management .....	45
11.1	Detailed Schedule.....	45
11.2	Budget .....	45
11.3	Risk Analysis .....	46
11.4	Localization Plan.....	47
11.5	Test Plan .....	47
11.5.1	Test Strategy.....	47
11.5.2	Test Schedule .....	47
12	Appendices .....	50
12.1	Project Team Description .....	50
12.1.1	Programming.....	50
12.1.2	Art.....	50
12.2	Asset List.....	51
12.2.1	Art.....	51
12.2.1.1	Model and Texture List.....	51
12.2.1.2	Animation List.....	51
12.2.1.3	Effects List.....	51
12.2.1.4	Interface Art List.....	51
12.2.1.5	Cut scene List.....	51
12.2.2	Sound.....	51
12.2.2.1	Environmental Sounds .....	51
12.2.2.2	Weapon Sounds.....	51
12.2.2.3	Interface Sounds.....	51
12.2.3	Music .....	51
12.2.3.1	Ambient .....	51
12.2.3.2	“Action” .....	51
12.2.3.3	Victory .....	51
12.2.3.4	Defeat .....	51
12.2.4	Voice.....	51
12.2.4.1	Actor #1 lines.....	51
12.2.4.2	Actor #2 lines.....	51
12.2.4.3	Etc.....	51

# 1 Design History

<b>Date</b>	<b>Revision</b>	<b>Description</b>
February 06, 2008	1.1	Initial Design Document
March 31, 2008	1.2	Update Schedule and Testing
March 31, 2008	1.3	Complete Management, Testing, and Technical sections
March 31, 2008	1.4	Fix some typos
April 01, 2008	1.5	Compress all pictures to web quality
April 24, 2008	1.6	Add test cases
April 24, 2008	1.7	Add weapons and menus

## 2 Game Overview

- Fast-paced, frantic, arcade like game play.
  - Competitive carnage, violence with some humor.
- Action-oriented
- A mixture of human and AI in multiplayer.

### 2.1 Game Concept

1-4 Players go on a bug hunt to determine who is the best bug exterminator in the galaxy. Players will see who can kill the most computer controlled bugs in a set amount of time.

### 2.2 Feature Set

Wide open map to explore, ruthless bug enemies, distinct weapons to use.

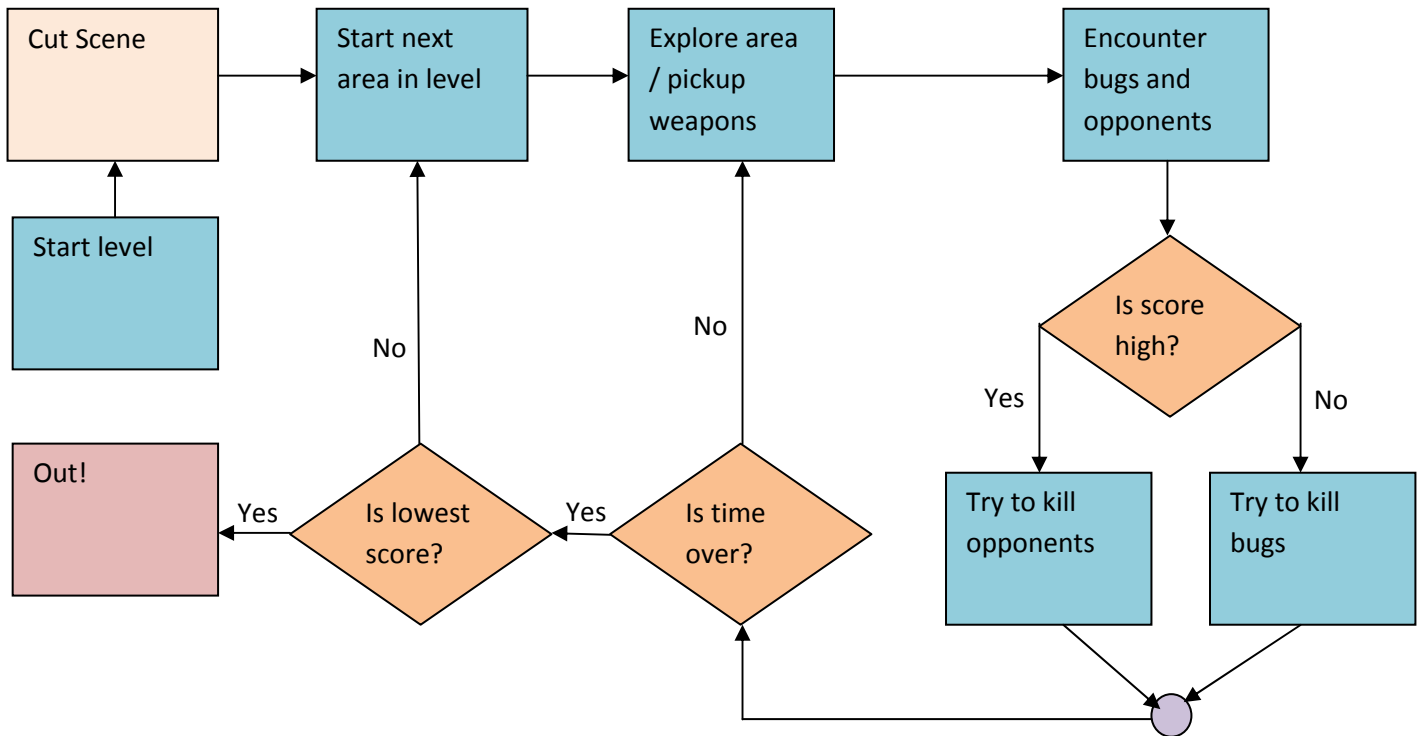
### 2.3 Genre

The closest genre to this game is the first person shooter / multiplayer genre with a sci-fi / star wars game look and feel.

### 2.4 Target Audience

The target audience is anyone who is interested in the CIS game festival. A wider audience may be people who love the star wars / sci-fi genre. This game would probably appeal more to people who are 13 years of age or older since it has some violence.

## 2.5 Game Flow Summary

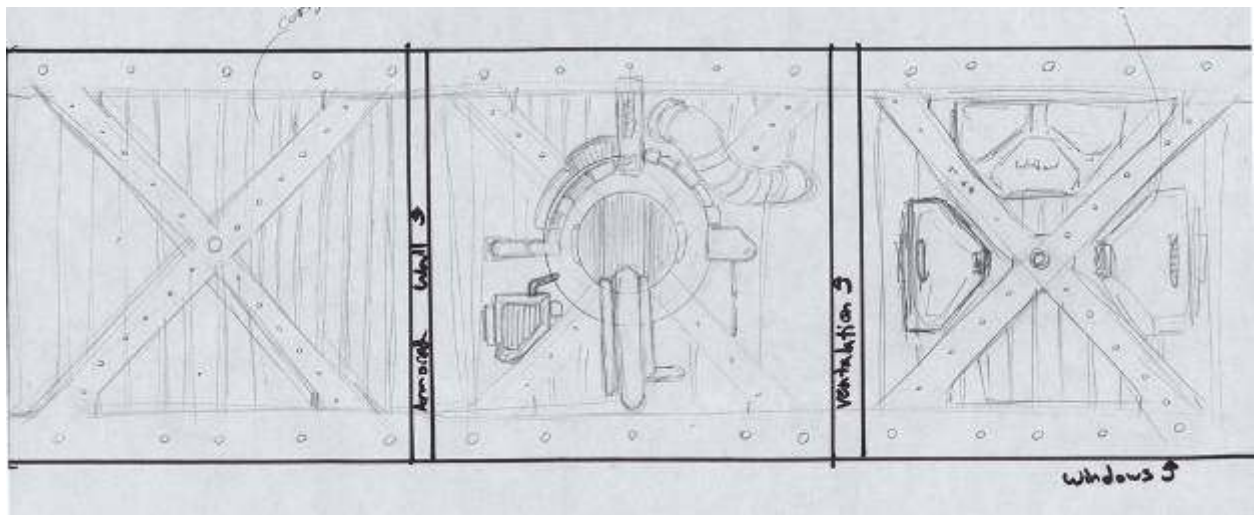
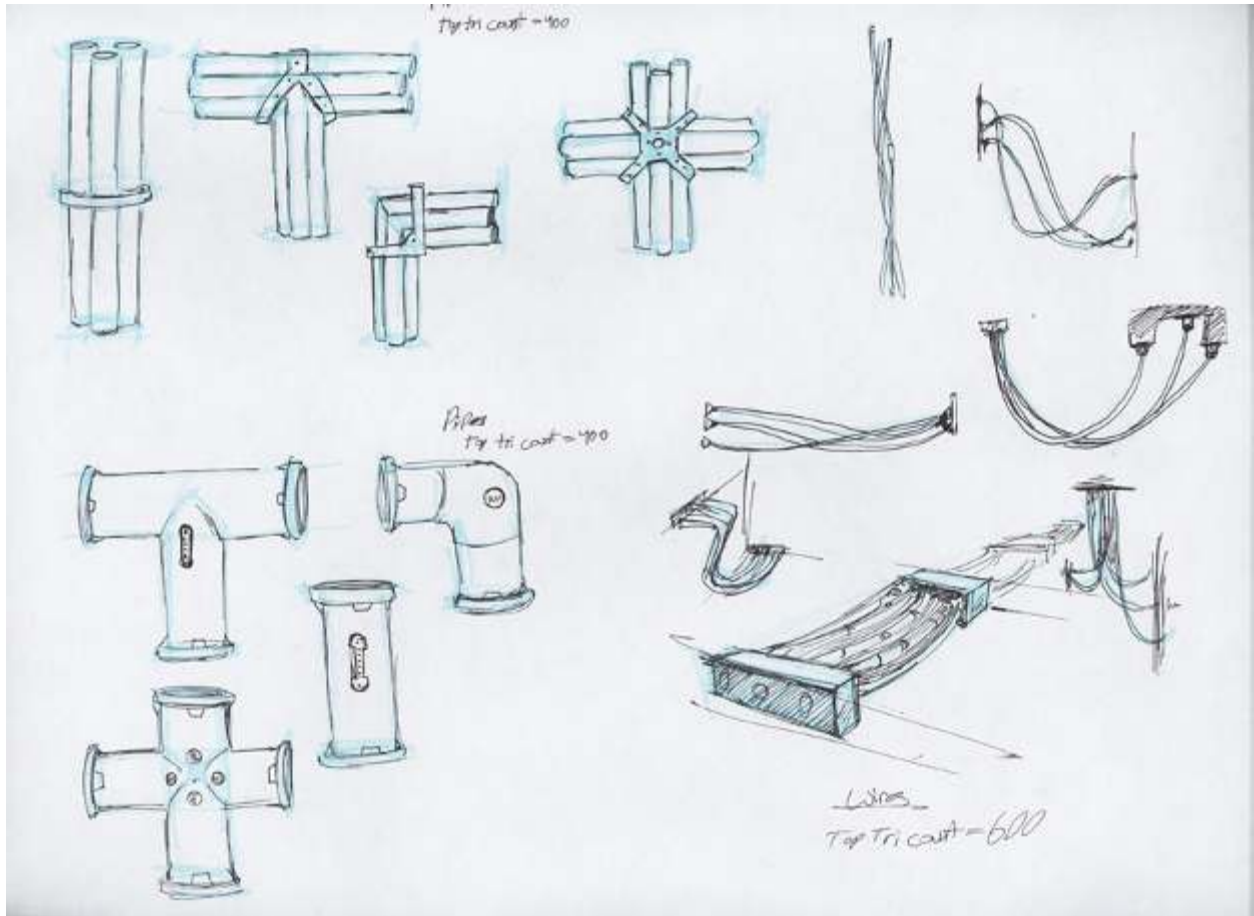


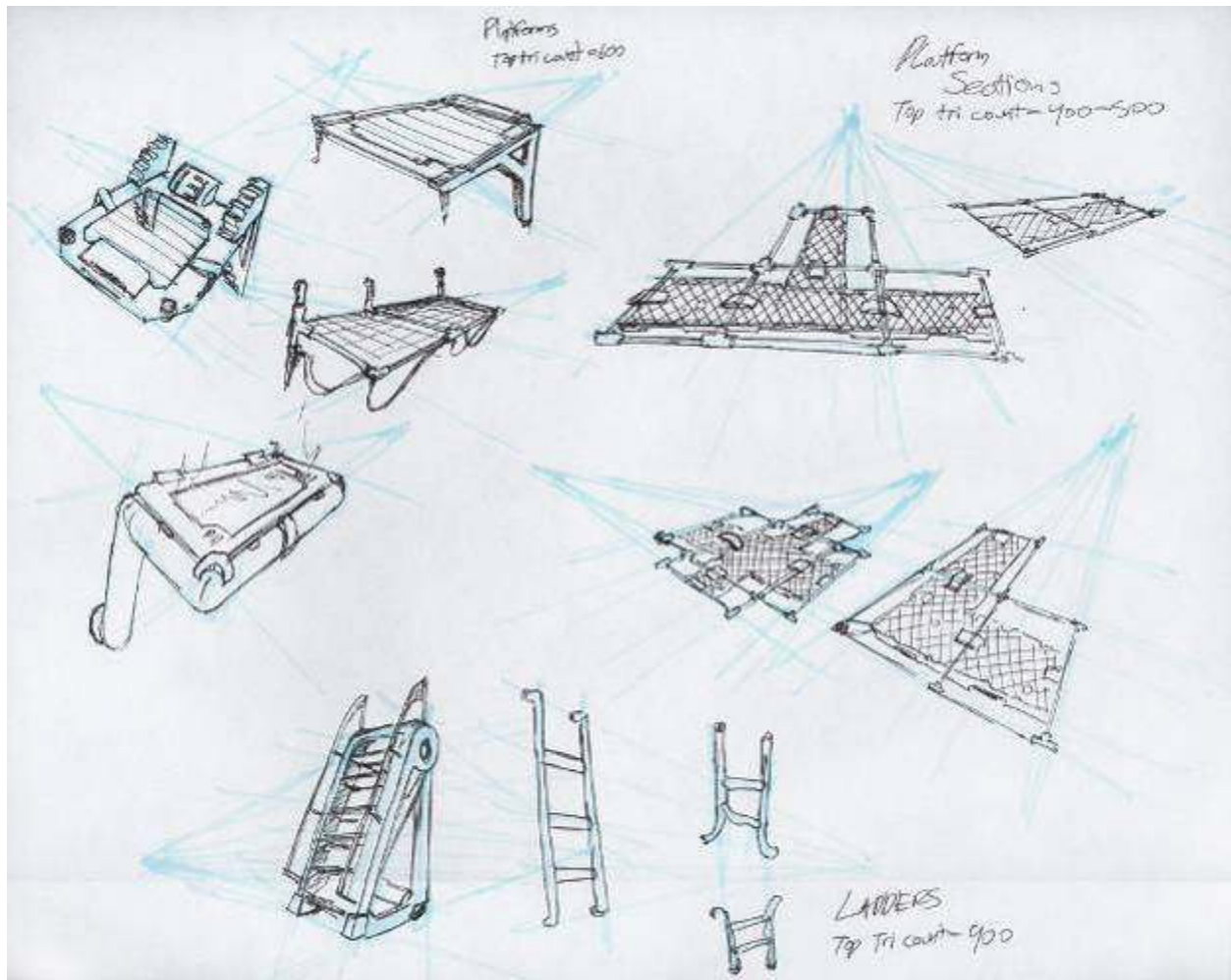
## 2.6 Look and Feel

What is the basic look and feel of the game? What is the visual style?

- First Person
- User interface display
  - Health meter
  - Current weapon and its remaining ammo
  - Current power-up, if any
  - Enemy kill count
- Inside an infested space station







## 2.7 Project Scope

### 2.7.1 Number of locations

The entire game takes place on a space station that has been infested by space bugs.

### 2.7.2 Number of levels

There is only one level, but it is sufficiently large to allow for a great deal of replayability

### 2.7.3 Number of NPC's

There is no set number of NPCs, as many bugs will be spawning throughout the game at different locations, based on certain factors within the game.

## 2.7.4 Number of weapons

There are 3 weapons with art/models and effects that are different than the weapons found in Unreal.

### 2.7.4.1 *Tranquilizer Rifle*

- Shoots “needles”
- This is the main Space Exterminator rifle
- Use existing Lightng Gun fire until correct art is supplied



### 2.7.4.2 *Sticky Gun*

- Similar to “Bio Rifle”
- It shoots liquid / goop
- Use existing “Bio Rifle” fire



### 2.7.4.3 *Engine Gun*

- Similar to “minigun”
- It shoots bullets as first fire and grenades as second fire
- Use existing “minigun” first fire and “Assault Rifle” second fire



## 3 Game Play and Mechanics

- The player must destroy as many insectoid infestations as possible before the other players do.
  - Enemies attack in swarms, are weak, and earn you points if killed.
  - Designed for fast-paced, arcade-style action
- After so many enemies have been killed, a door will open, allowing all but the lowest scoring player to enter, and so on and so forth.
- The lowest scoring player in each round will not respawn. The last man left in the final stage win.
  
- Other opponents will try and kill more first!
- The bugs bite!
- Your weapon only has so much ammo!
  
- You can kill your opponent, though respawn time is fast, and bugs make more points.
- Armor and Health are hidden throughout the station
- Bigger and more powerful extermination weapons are also abundant.

### 3.1 Gameplay

#### 3.1.1 Game Progression

The game will advance to the next level after a certain amount of enemies have been killed. However, the player with the lowest score will be eliminated and will not advance. The game will continue until all but one player has been eliminated, this player being the winner.

#### 3.1.2 Mission/challenge Structure

The goal in each level is to not be the player with the lowest score. A player's score can be increased by killing the insectoid enemies. It may also be possible to slow down other players by killing them. In each level, there may be items that can be picked up to make the players character more powerful, including, armor, health, and weapons.

#### 3.1.3 Objectives

The objective of the game is to win by never having the lowest score in any level. Or, to score as high as possible.

#### 3.1.4 Play Flow

The player will enter the game and the match will begin when all players have entered. Players will immediately set out to find bugs to kill, and may attack other players if they come across them. The player will have to move from one point to another in order to find the most bugs, as they will not

immediately respawn at a given location. When the match is over, the player with the highest score is declared the winner.

## 3.2 Mechanics

What are the rules to the game, both implicit and explicit? This is the model of the universe that the game works under. Think of it as a simulation of a world, how do all the pieces interact? This actually can be a very large section.

## 3.3 Physics

The game will use all of Unreal's default physics

### 3.3.1 Movement

#### *3.3.1.1 General Movement*

Players and bugs will move by walking/running around the map toward their destination. The bugs will be slower than the player so that the player will be able to run away from a sticky situation. The players will not move any faster or slower than other players can move in other Unreal games.

#### *3.3.1.2 Other Movement*

There will be areas of the map where players can teleport or take elevators from one place to another in order to get to a location faster. These shortcuts will not be immediately noticeable to a player, each player will have to discover them for themselves.

### 3.3.2 Objects

#### *3.3.2.1 Picking Up Objects*

The only objects in the game will be the ammo and health pickups that are strewn about the map, they can be picked up by walking over them just like in any other Unreal game.

### 3.3.3 Actions

#### *3.3.3.1 Switches and Buttons*

Some elevators will require that the player press a switch or button to activate

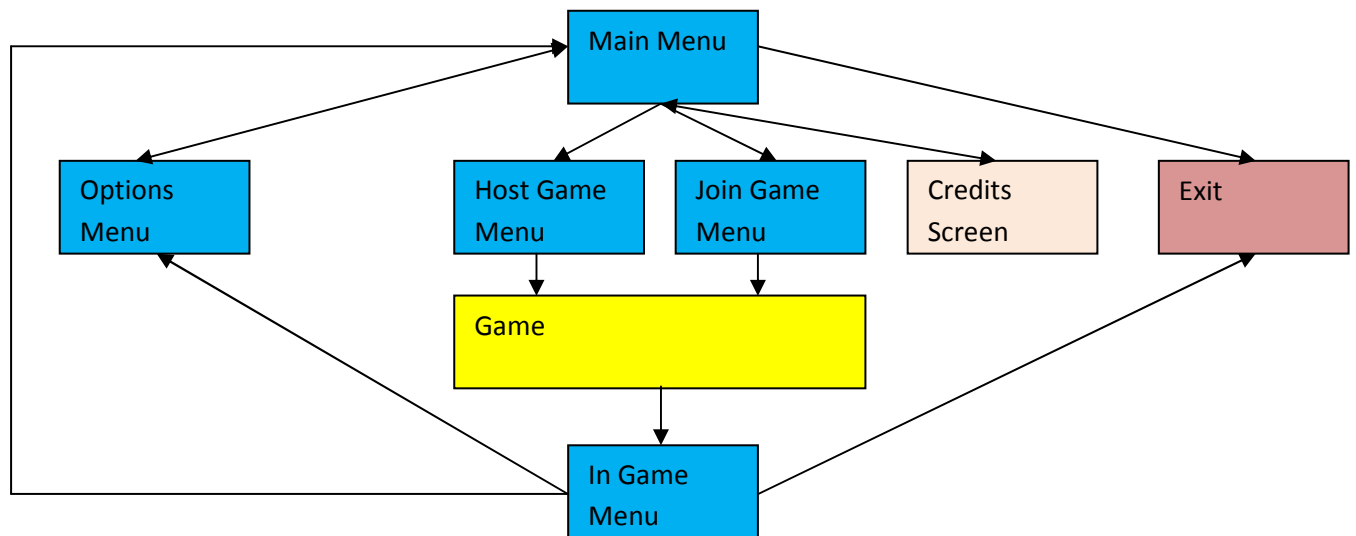
### 3.3.4 Combat

All player combat will be performed with the use of ranged weapons (i.e. guns), this includes players fighting enemies and players fighting players. Each weapon will have different projectile types, firing

rates, damage, and ammo storage/usage. The AI controlled bugs will be using melee attacks only, attacking with their claws and fangs.

## 3.4 Screen Flow

### 3.4.1 Screen Flow Chart



### 3.4.2 Screen Descriptions

#### 3.4.2.1 Loading Screen





### 3.4.2.2 Main Menu Screen

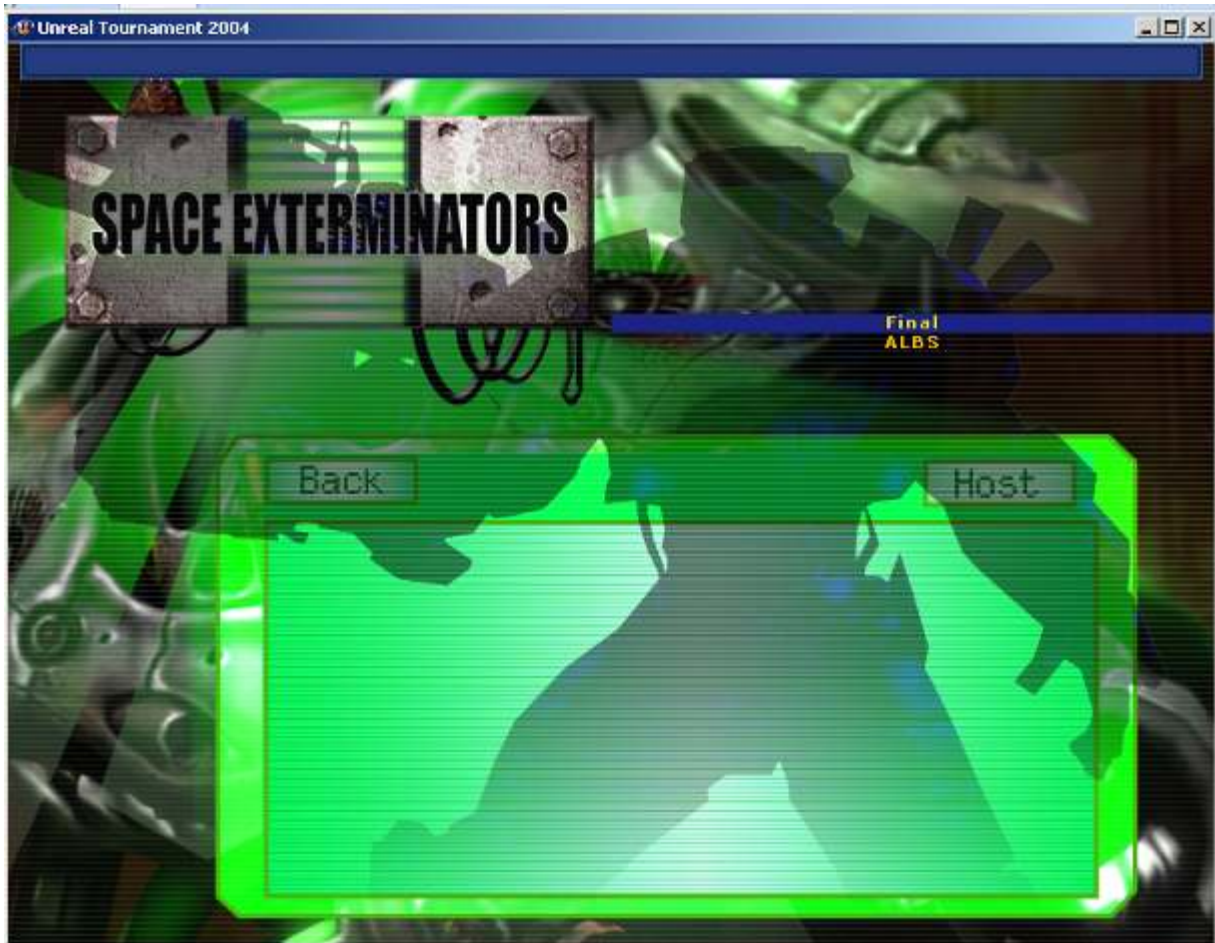
Below are some concept menu screens.



Below is the actual main menu screen.



### 3.4.2.3 Host Screen



## **3.5 Game Options**

The game will use the same options as the base Unreal Tournament 2004 game, which includes different settings for input, graphics, audio, and network.

## **3.6 Replaying and Saving**

## **3.7 Cheats and Easter Eggs**

## 4 Story, Setting and Character

### 4.1 Story and Narrative

#### 4.1.1 Back story

In an area of space that is well known throughout the galaxy for bug infestation there exists a space station that all the greatest exterminators call home. At any given time there may only be a handful of exterminators on the station, the rest of them being out on assignment, but today is a special day....

A 'bug gold rush' of sorts is dying down, and now there are too many exterminators in the area, causing business to slow for everyone. The previously busy exterminators are left with a lot of free time, and in space the only thing left to do is drink and cause trouble.

Exterminators sit and discuss their exploits, all while drinking the bar dry. This combination leads to heated arguments and false bravado. The exterminators begin arguing over which one of them is the best in the galaxy, with many of them throwing false threats and smack talk, never thinking they'd have to back it up...

But just then, the station alarms sound, there is a hull breach in the lower cargo decks. Commotion overruns the station as it is learned that the bugs have taken residence right in the exterminator's backyard, right under their noses. Now is the time to put up or shut up. Quickly a bet is made to see who can kill the most bugs, money is pooled and it's winner take all.

One exterminator will win himself enough money to stay drunk for a month, who will it be?

#### 4.1.2 Plot Elements

The plot will be explained and/or shown to the player prior to the game beginning so that they know what exactly the story is. At the end of the game the results will be tallied and one player will win the bet, and the game. The story will likely not play a major role throughout the actual gameplay.

#### 4.1.3 Game Progression

The player will choose a character and enter the game as that character. They will play through the game as that character, and at the end one of the players will have won the game with that character.

#### 4.1.4 License Considerations

#### 4.1.5 Cut Scenes

##### 4.1.5.1 Cut scene #1

Initial setup of the story, explaining where the story takes place, introducing the characters involved, and setting up for the events of the game.

#### *4.1.5.1.1 Actors*

Every character in the game will appear in the first cut scene.

#### *4.1.5.1.2 Description*

The station will be shown and explained, along with the situation that the characters find themselves in. The characters will be shown at the bar drinking, and then start arguing. The alarms will go off, the bet will be made, and the exterminators will head out to infested area, where the game begins.

#### *4.1.5.1.3 Storyboard*



#### *4.1.5.1.4 Script*

### *4.1.5.2 Cut scene #2*

#### *4.1.5.2.1 Actors*

#### *4.1.5.2.2 Description*

#### *4.1.5.2.3 Storyboard*

#### *4.1.5.2.4 Script*

## **4.2 Game World**

### **4.2.1 General look and feel of world**

- Inside an infested space station

## 4.2.2 Area #1

*4.2.2.1 General Description*

*4.2.2.2 Physical Characteristics*

*4.2.2.3 Levels that use area*

*4.2.2.4 Connections to other areas*

## 4.2.3 Area #2

*4.2.3.1 General Description*

*4.2.3.2 Physical Characteristics*

*4.2.3.3 Levels that use area*

*4.2.3.4 Connections to other areas*

## 4.3 Characters

### 4.3.1 Character 1: Earl

#### *4.3.1.1 Back story*

Born on an off-world moon colony, Earl had dreams of being a space marine, but could never get sober enough to do it. Instead Earl took up his fathers calling of bug extermination, which doesn't afford him a lavish lifestyle, but does pay for the bills...and the booze. Now in his mid 30s, Earl's goals haven't changed in over a decade, for him the status quo is the only life he's known.

#### *4.3.1.2 Personality*

Although he'd never be considered a model student or citizen, Earl is actually fairly bright, more so than people give him credit for. Even though he doesn't fully understand most things, he has a knack for getting the job done his own way, and it works out more often than not. Earl enjoys the satisfaction of a job well done, and he rewards himself highly for it, he blows most of his pay the night that he receives it, and thus is always looking for the next big score. Earl's got a real competitive streak which grows exponentially when he gets some liquid courage in him. That combination of ego and booze is what landed him in this contest, and he'll need to use it to emerge victorious.

#### *4.3.1.3 Look*

##### *4.3.1.3.1 Physical characteristics*

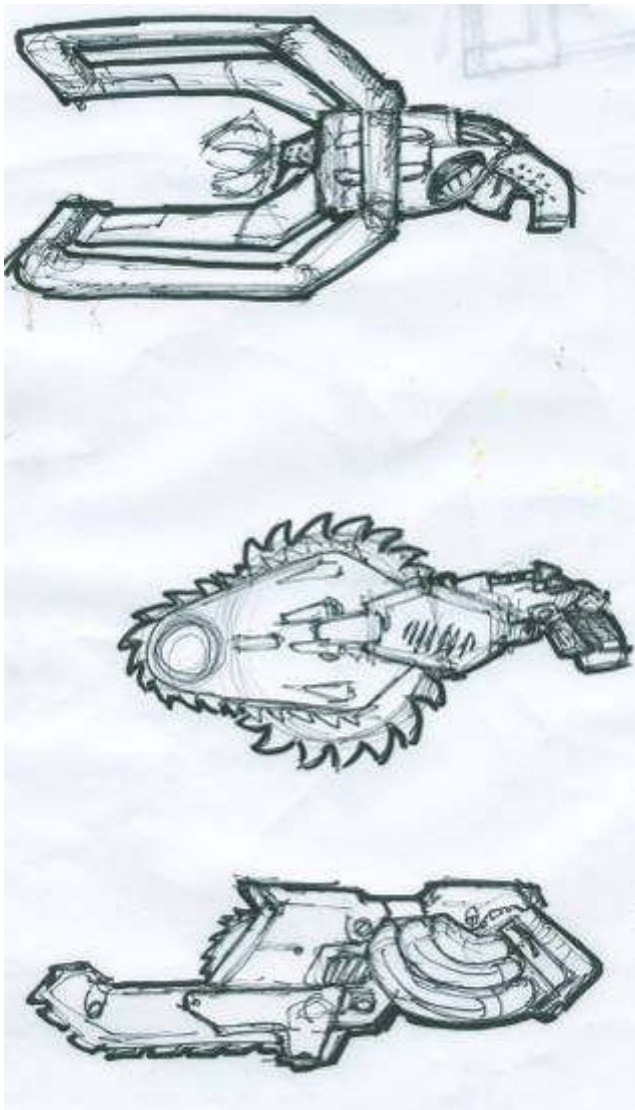
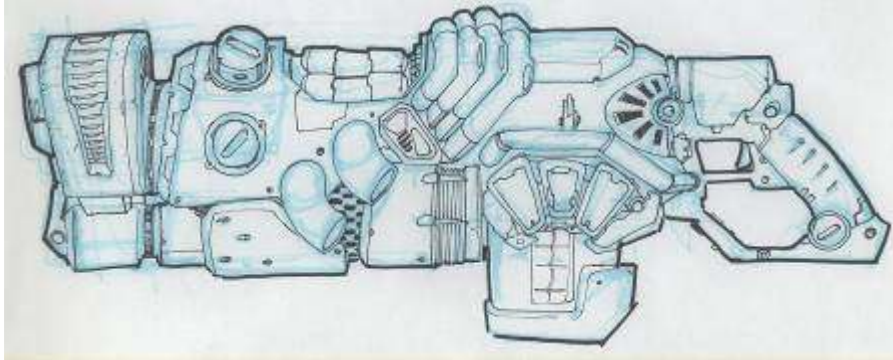




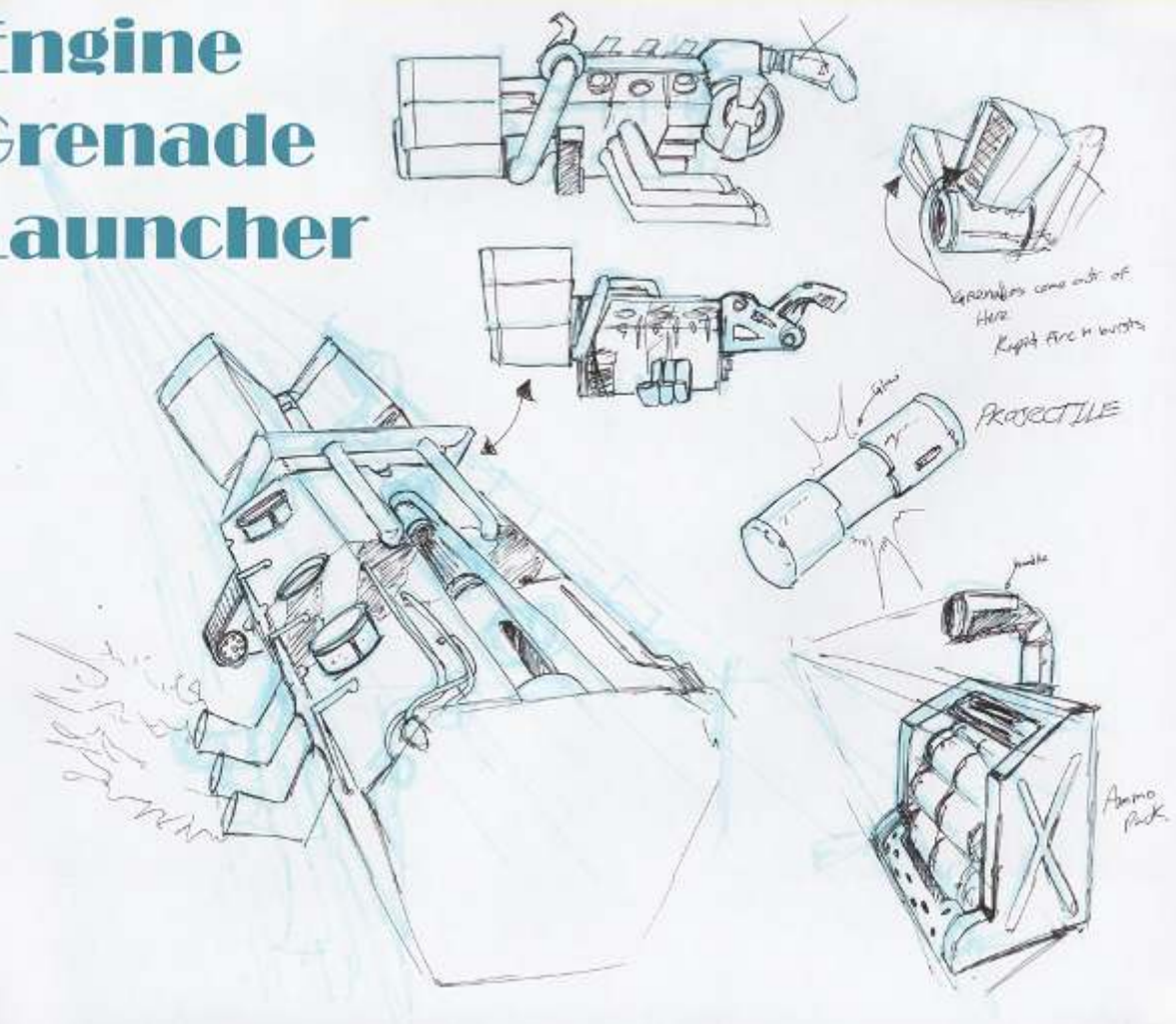
4.3.1.3.2 Animations

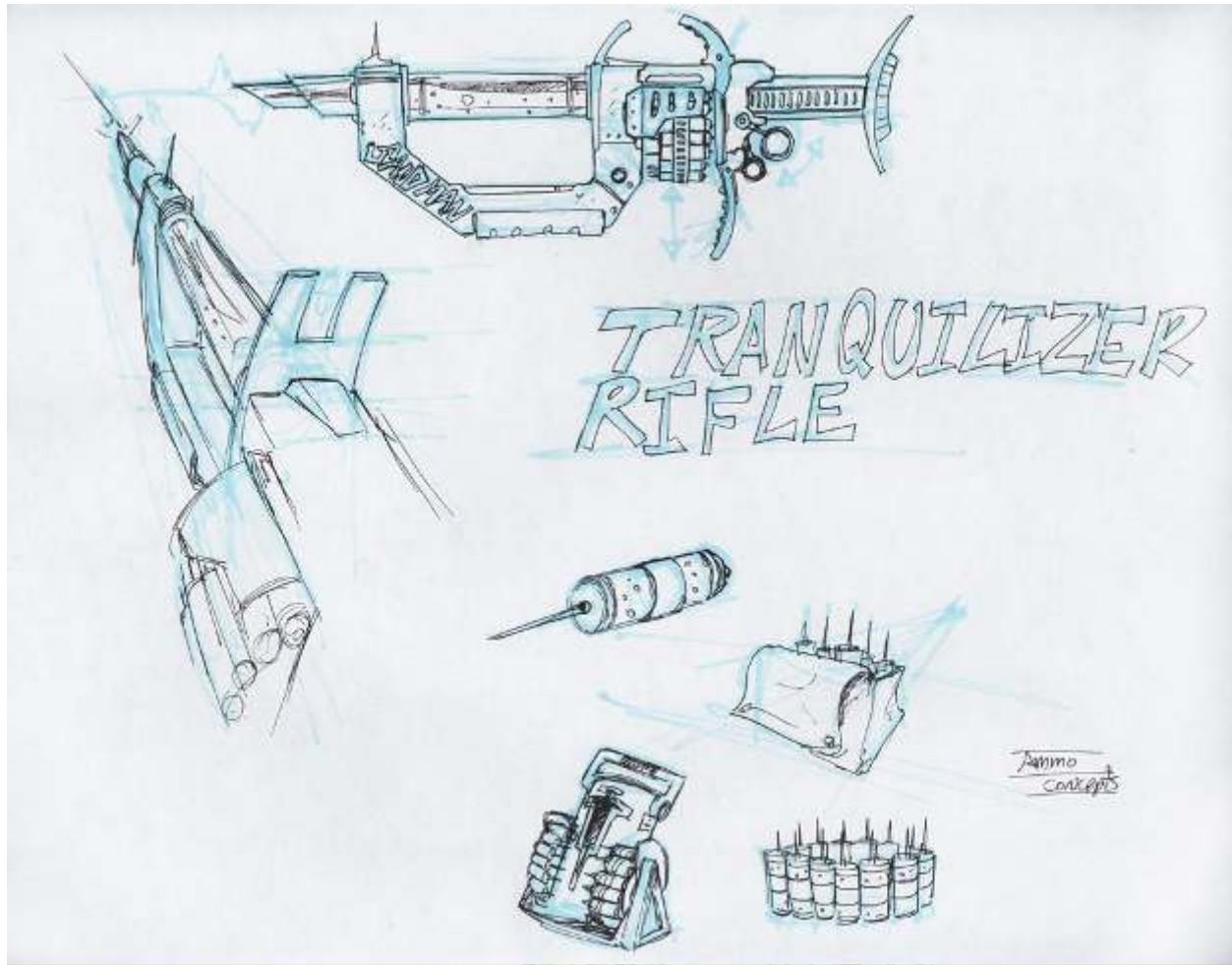
#### 4.3.1.4 Special Abilities

Earl is a master with his bug extermination equipment. A true professional, he is well versed with many cutting edge anti-bug weapons.



# Engine Grenade Launcher





#### *4.3.1.5 Relevance to game story*

Earl is the main character in the game story.

#### *4.3.1.6 Relationship to other characters*

Earl is Bugg's competitor.

#### *4.3.1.7 Statistics*

Age: Mid 30's

Height: 6' 2"

Weight: 255 lbs

Blood Alcohol Level: 0.04 (average)

Ammo rounds on person: 450

## 4.3.2 Character 2: Bugg

### 4.3.2.1 Back story

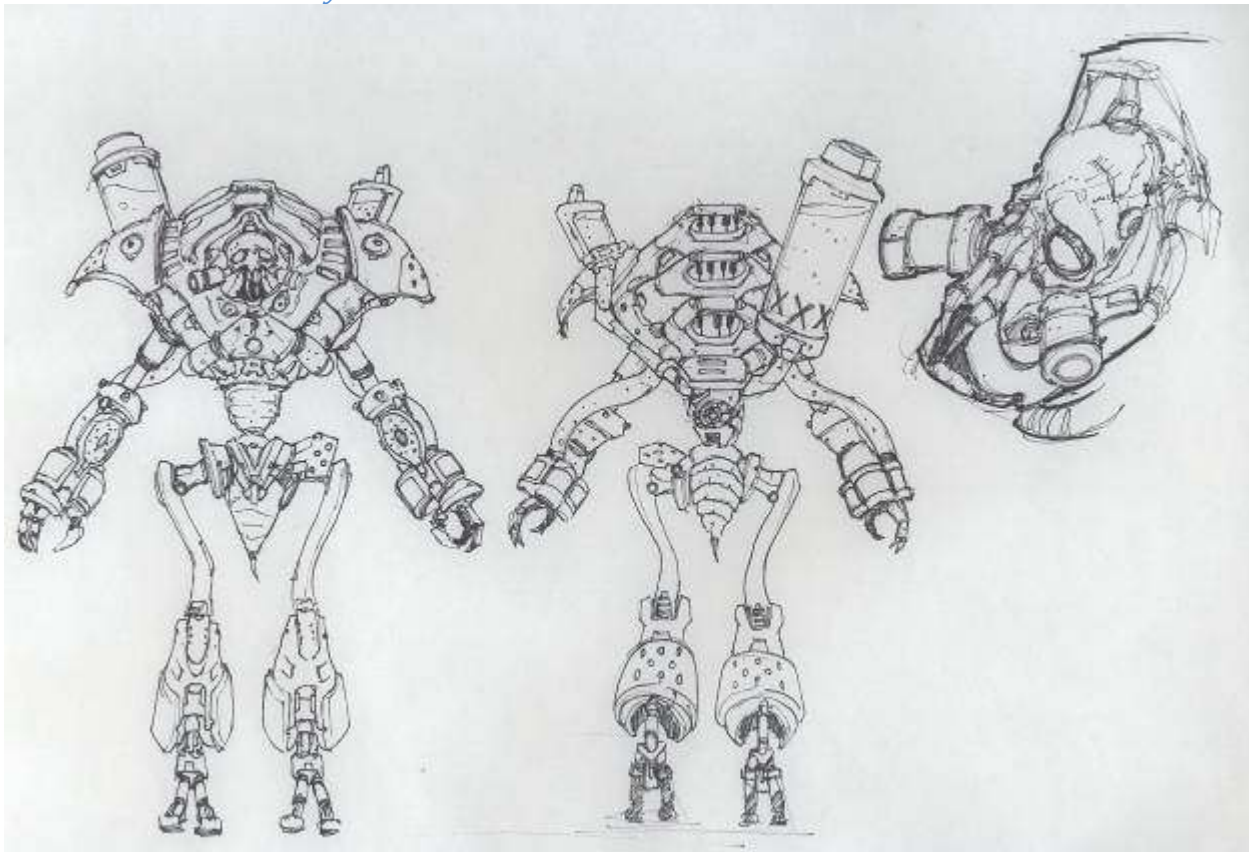
If there's one sentient being in the universe that drinks more than Earl, it's Bugg. A genetically enhanced humanoid bug, he may actually be a relative of the very same bugs that he hunts, but he couldn't care long enough to find out. His body is extremely frail, which is why he spends his life in a power suit that augments his abilities. Bugg's origins are mysterious, but it is most likely that he was some kind of experiment gone wrong. Whatever his past is, his present is perfectly clear, a cold blooded exterminator.

### 4.3.2.2 Personality

Bugg has an innate distrust and dislike of humans, unless of course they're willing to pay him. Exterminating is the only thing that Bugg knows and it's the only thing that he does. With no friends or family to speak of, Bugg is free to work 18 hour days and spend the other 6 hours drinking. Bugg is very irrational and unpredictable, and he also has a need to stand out, which is what drew him into this competition. He is liable to go to any lengths to achieve his goals, no matter how under-handed they may seem. His power suit is full of tricks, and he makes great use of all of them.

### 4.3.2.3 Look

#### 4.3.2.3.1 Physical characteristics





Height: 6' 5"

Weight: 65 lbs (330 lbs in power suit)

Power suit level: Over 9000!

## 5 Levels

### 5.1 Level #1

#### 5.1.1 Synopsis

The Game will consist of one main level that will be extended until the game mission is completed. Up to four players begin playing on the level in different locations while the “enemies” appear at random throughout. As the players search the level, they encounter the “enemies” and must kill them to gain points and not to be killed themselves. After a certain number of “enemies” are killed, the level will expand to a new area while the old area will no longer be accessible. The new area will still be within the ship that the players are in but will be another section of it.

#### 5.1.2 Introductory Material



Illustration 5.1 – Introductory Story Line (Inside the Ship)

The level will begin with a short introduction to the game. The player will hear the story that leads up to the mission. There will also be a visual to get the player more familiar to what has taken place before



the game began. This introductory material can be suppressed by the player if he or she wishes to skip it.

### 5.1.3 Objectives

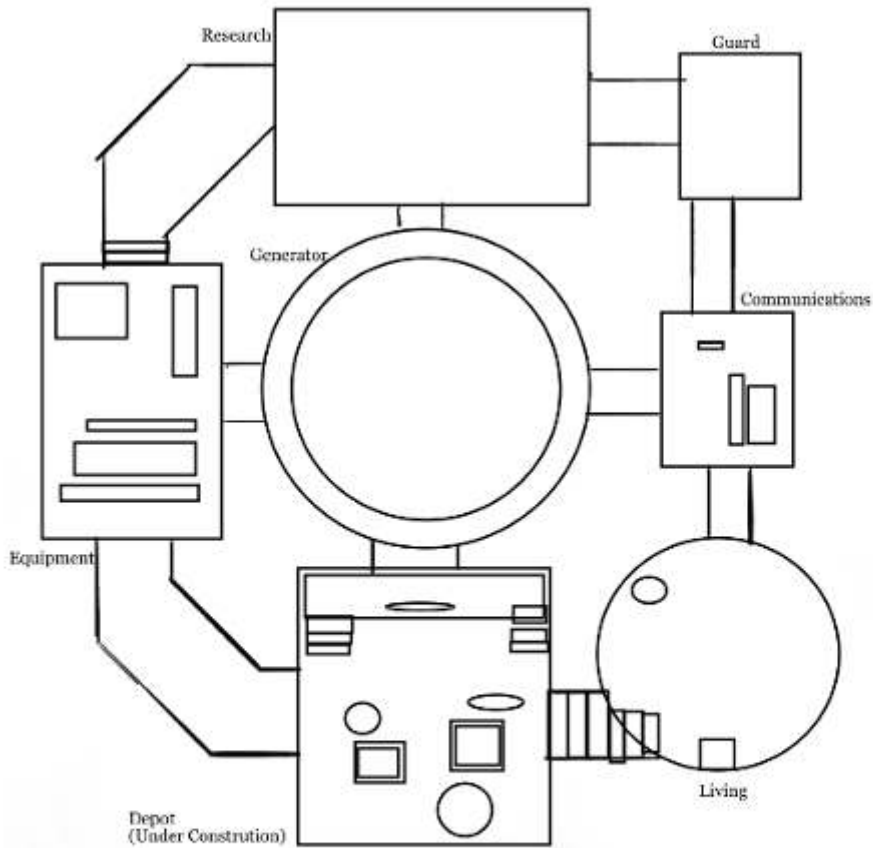
The objectives of the level are the main game objectives. Players compete to kill as many “enemies” as possible. Once a certain number of “enemies” are killed, all players except the lowest scoring player will move on to the next area.

### 5.1.4 Physical Description

Similar to illustration 5.1, the level will be within a space center. There will be wreckage and destruction from what the bugs have created. The lighting will be dim and spotty with the interior of the ship being dark metal, giving a dingy appearance. Some areas will be spacious while others being narrow and crowded.

### 5.1.5 Map

As shown in the below illustration the level consists of mainly open space with obstacles. There are also areas that have entrances (indicated by a protruded side) where a player can enter.



# SPACE EXTERMINATORS

Illustration 5.1.2 – Level Layout

## 5.1.6 Critical Path

There are no critical paths within the level.

## 5.1.7 Encounters

The player will encounter “enemies” at random locations throughout the map. The player may also run into other players.

## 5.1.8 Level Walkthrough

The level is open to exploration and there is no real reason to traverse a particular path within in. There are also no triggers within the level that cause anything to happen.

## 5.1.9 Closing Material

Once the level is completed, by the killing of a specific numbers of “enemies”, the players will be notified and will then lose control of their unit. The level will then be replaced with a new variation of it, giving the effect of a new level. The player with the lowest score will be notified and removed from the game. In the event of a tie, all players would continue to the next level. If there is only one player advancing then the game would be over and the player congratulated.

## 5.2 Level #2

### 5.2.1 Synopsis

This level is a continuation of level #1 with random variations of the layout of obstacles. The players that have made it this far, by not having the lowest score, will battle it out again to get the most kills on the enemy units.

### 5.2.2 Introductory Material

There will be no introductory material.

### 5.2.3 Objectives

The objectives are the same as level #1.

### 5.2.4 Physical Description

The same as level #1

### 5.2.5 Map

The map will be similar to level #1 with slight variations of the location of obstacles and enemies.

### 5.2.6 Critical Path

There are no critical paths within the level.

### 5.2.7 Encounters

The player will encounter “enemies” at random locations throughout the map. The player may also run into other players. This is the same as the previous level

## 5.2.8 Level Walkthrough

The level requires no walkthrough as explained in level #1

## 5.2.9 Closing Material

The closing material will be similar to level #1. If only three players were playing and there was no tie, then the game would end congratulating the player with the highest score. If there are four players or a tie with three players, then the game would go on for another round (the details of this next level would be the same as level #2).

## 6 Interface

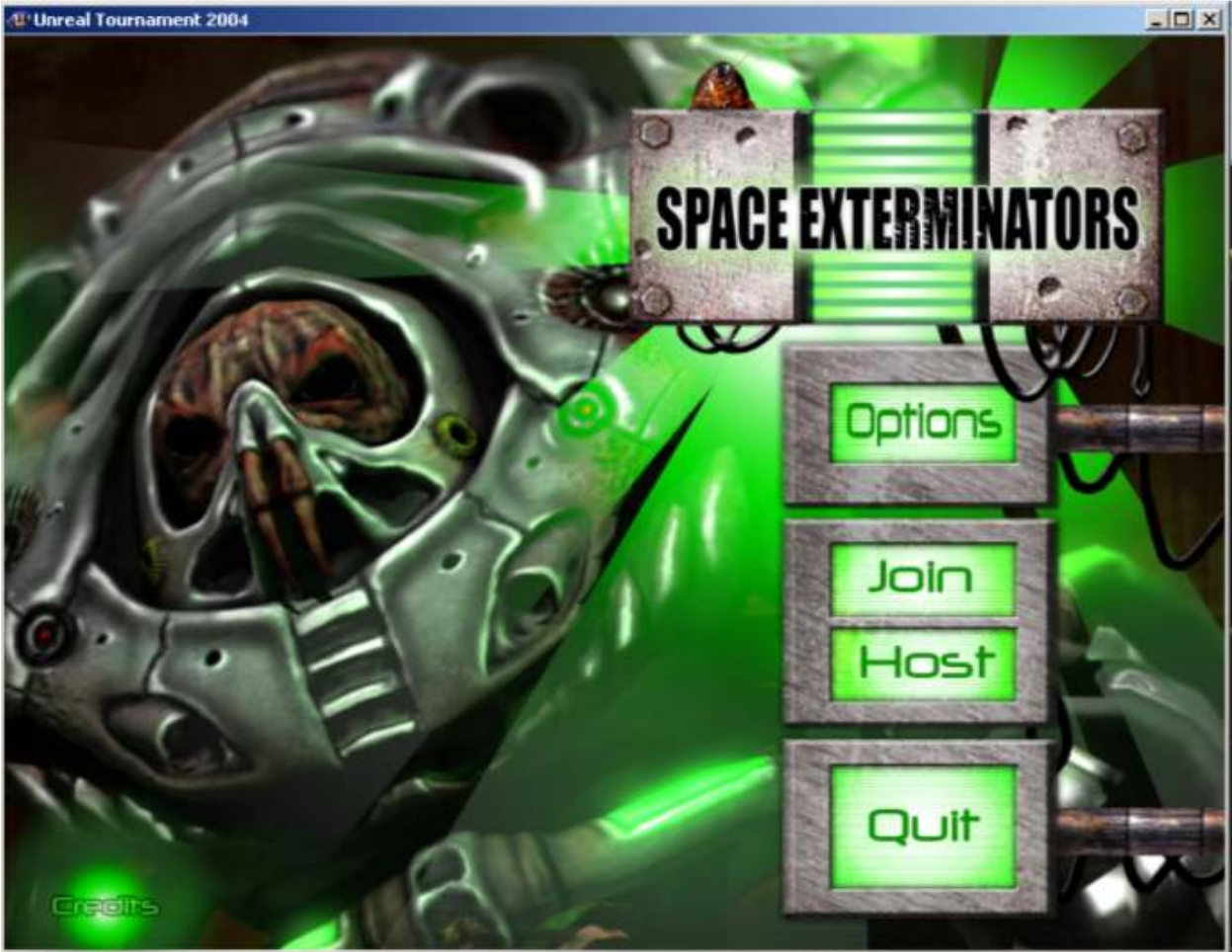
### 6.1 Visual System

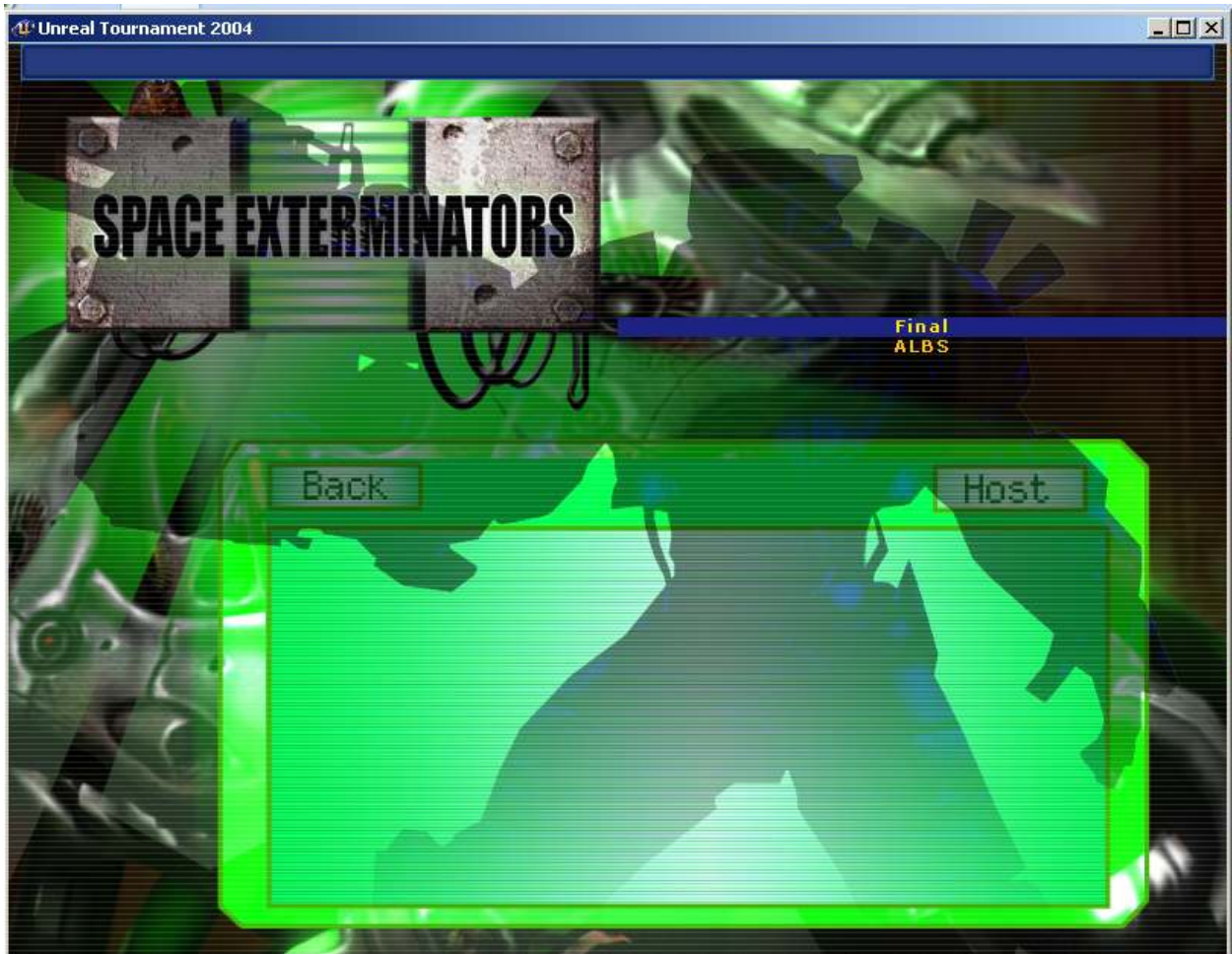
#### 6.1.1 HUD - What controls

- User interface display
  - Score
  - Ammo
  - Health
  - Weapons



6.1.2 Menus





### 6.1.3 Rendering System

We will use the basic Unreal Tournament 2004 rendering system.

### 6.1.4 Camera

We will use the basic Unreal Tournament 2004 camera control system.

### 6.1.5 Lighting Models

We will use the basic Unreal Tournament 2004 lighting models.

## **6.2 Control System**

We will use the basic Unreal Tournament 2004 control system.

## **6.3 Audio**

We will use the basic Unreal Tournament 2004 audio.

## **6.4 Music**

We will use the basic Unreal Tournament 2004 music.

## **6.5 Sound Effects**

We will use the basic Unreal Tournament 2004 sound effects.

## **6.6 Help System**

A Help system is not provided.



# 7 Artificial Intelligence

## 7.1 Opponent AI

Since this game is primarily player vs. enemy, computer AI plays a major part. The built in AI of Unreal will handle most of the basic details, and we will fill in the rest.

## 7.2 Enemy AI – Villains and Monsters

The main opponent of the player are the bugs that have infested the station, therefore the computer AI should have a bug/hive-mind mentality. They stick together, stay close to 'home', and viciously attack any intruders.

The bugs will spawn at various points of the map, and they will stick around the area that they spawned in, patrolling it for intruders. They should not leave the area and actively search for players to fight, but they will defend their territory to the death.

The bugs are derived from the Unreal Monster class (which is derived from Pawn, the same thing the Bots are derived from), this class provides us with the basic seek and fighting AI that the bugs will use. From there we will tweak the little things like sight and hearing distance, speed, health, damage. One single bug won't be difficult to kill, but going into a room carelessly and getting surrounded by 10 of them will not be an easy situation to get out of.

Spawning is handled by a dummy pawn which acts as sort of a Monster factory, it sits hidden and calls a ScriptedSequence to spawn the bugs and send them on their way. A monster factory should spawn a set amount of bugs for a given area, when the areas has been wiped out it will spawn another set after a given amount of time. This ensures that a player cannot sit and camp one area to rack up easy points on bug spawns.

### 7.2.1 Player and Collision Detection

Collision detection will be handled by Unreal using its default collision detection. We have reduced the size of the collision cylinder for the bugs so that a player can jump over them, which may be the only escape if a player is surrounded by bugs.

### 7.2.2 Pathfinding

Pathfinding will be handled by Unreal using path nodes and other map actors to set up a navigation system for each different set of bugs. Each area that a bug spawns in will have a pathing network to keep the bugs from leaving the area, with the exception of bugs that spawn from triggers, which will immediately attack a player.

## 8 Technical

### 8.1 Target Hardware

The target hardware requirements are the same as Unreal Tournament 2004 requirements.

---

	Minimum	Recommended
<b>Processor</b>	1.0 GHz	1.2 GHz
<b>Memory</b>	128 MB	256 MB
<b>Hard Drive</b>	6.5 GB free	
<b>DVD-ROM</b>	6X	
<b>Video</b>	32 MB	64 MB NVIDIA Geforce2

---

### 8.2 Development hardware and software

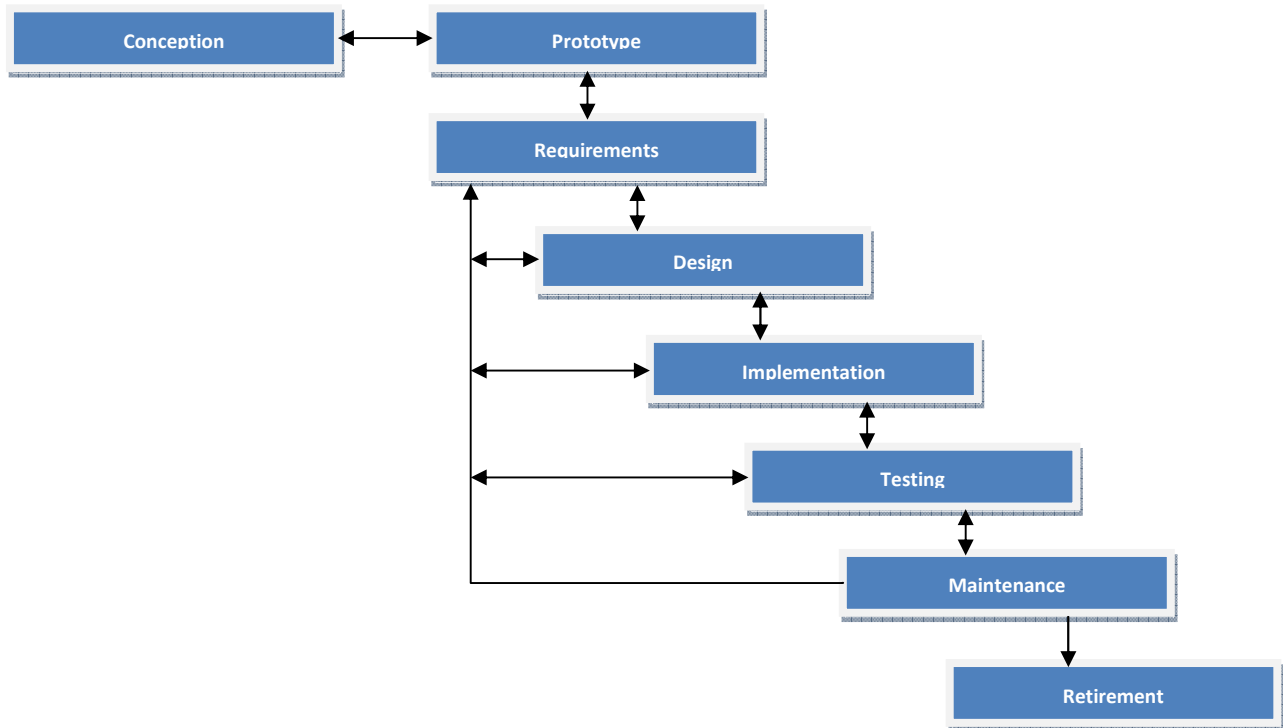
For the programming development, the hardware requirements are the same as the target hardware. See section “8.1 Target Hardware” for more details. The software requirements are the same as Unreal Tournament 2004 requirements, and those are:

- Unreal Tournament 2004
- Windows 98/Me/2000/XP
- DirectX 9.0b

The art development requirements may be a little different. The required software may include different art content creation tools such as Autodesk Maya, Milkshape, and other modeling software packages. The required hardware may include hardware that is capable of running such software packages. However, from a programming development standpoint, the product of the art team will be treated as third-party components.

## 8.3 Development procedures and standards

Our organization will follow the Chief Programmer structure. We will follow the Rapid Prototype lifecycle model for our project lifecycle. The diagram below shows an outline of the Rapid Prototype lifecycle model. Early on in the project, we built a prototype of the SpaceX mod which we used to formalize our specifications.



Our standards include specification reviews and code reviews to ensure a lower defect density and better product quality. Our development team members are expected and encouraged to review the other members' code to verify its correctness. We will be performing many types of tests to ensure proper software behavior.

## 8.4 Game Engine

We will use the Unreal Tournament 2004 game engine (including UnrealEd).

## 8.5 Network

Unreal Tournament 2004 has built in support for multiplayer games. We will just leverage this functionality.

## 8.6 Scripting Language

Unreal Script is the main scripting language used for the Unreal Tournament 2004 game engine.

## **9 Game Art**

This may be abbreviated with most of the content in an Art Bible.

### **9.1 Concept Art**

### **9.2 Style Guides**

### **9.3 Characters**

### **9.4 Environments**

### **9.5 Equipment**

### **9.6 Cut scenes**

### **9.7 Miscellaneous**

## **10 Secondary Software**

### **10.1 Editor**

### **10.2 Installer**

### **10.3 Update software**

# 11 Management

## 11.1 Detailed Schedule

The table below outlines the project schedule. The project is expected to be completed within 3 months. There are **4 developers** assigned to the project, and each is expected to contribute about **6 hours/week** (expected homework for a 3 credit class). Based on that, the total development effort (excluding art) would be about 288 hours or **1.8 PM**. We expect a similar effort and schedule for art design.

Date	Millstone Reached
1/28/08	Project conception: <ul style="list-style-type: none"><li>• Assign team roles</li><li>• Produce draft plan</li></ul>
2/11/08	Initial development starts: <ul style="list-style-type: none"><li>• Setup repository</li><li>• Build SpaceEx mod prototype</li><li>• Assign development components</li></ul>
3/03/08	Alpha Release <ul style="list-style-type: none"><li>• “Specification freeze” status</li></ul> Final development period <ul style="list-style-type: none"><li>• Complete functional requirements</li><li>• Fix bugs</li></ul>
4/07/08	Beta Release <ul style="list-style-type: none"><li>• “Code complete” status</li></ul>
4/23/08	Testing period <ul style="list-style-type: none"><li>• Initial acceptance tests</li><li>• Final bug fixes</li><li>• Regression tests</li></ul>
4/25/08	Gold Release

## 11.2 Budget

The table below shows the estimated costs of a similar project. However, since there are no development or art costs, then the real costs would be just those of purchasing UT2004 and the Mastering Unreal Technology book (not taking into account the time that we have to put into the project). Those costs total about **\$280.00**.

Item	Item Cost \$	Quantity	Total \$
Unreal Tournament 2004 [Editor’s Choice Edition DVD]	20.00	4	80.00
Mastering Unreal Technology book	50.00	4	200.00
Development hours	20.00	288	5,760.00
Art design hours	15.00	288	4,320.00
<b>Total</b>			<b>10,360.00</b>

## 11.3 Risk Analysis

The table below outlines the major risks involved with this project and the approach we took to encounter them. For the most part, we took a reactive approach by watching the risks closely and only getting involved when necessary.

Risk Id	Description	Category	Probability	Impact	Mitigation Method
RISK_01	Specification change	SPEC	70%	2	Discuss with team to freeze spec. early
RISK_02	Development schedule overrun	MGMT	30%	3	Assess which features could be removed
RISK_03	Team members leave / do not participate in project	MGMT	50%	3	Break project down into smaller components and only assign each component at a time
RISK_04	Art incompatibility (import / export issues)	ART	80%	1	Work with art team to ensure compatibility
RISK_05	Art content not useable (incorrect animations, etc.)	ART	20%	2	Familiarize art team with expected art content
RISK_06	Art schedule overrun	ART	80%	2	Treat art as a secondary feature and only use it when it is available

It is important to note that we have realized some of the risks, and thus our project is facing some hardships to have all of its features completed on time. For example, we have realized RISK\_03 very early when one of our development team members left the team. Also RISK\_02 is very likely to happen and we are now trying to think of feature to remove from the project.

## 11.4 Localization Plan

There is no plan to build any localized versions of the project.

## 11.5 Test Plan

### 11.5.1 Test Strategy

The team will take a proactive approach to testing. The members of the team are expected to unit test their components before committing changes. Furthermore, integration testing will take place on a regular basis to ensure that changes to one component do not break other components. The team members are encouraged to run the game at least once after making code changes to ensure proper behavior. Acceptance testing will take place near the end of the development project (after Beta) to catch any outstanding bugs prior to the Gold release.

In summary, the types of tests we will be performing are:

- Units tests: Prior to any code change
- Integration tests: After every code change
- Acceptance tests: After Beta Release and prior to Gold Release
- Regression testing: Whenever bugs are discovered, reported, and fixed, regression tests are run

### 11.5.2 Test Schedule

Acceptance tests will take place after Beta Release and prior to the Gold Release. All the team members are required to participate in acceptance testing and report any outstanding issues to their respected owners for fixes. It would be best to perform these tests on a black-box testing basis to ensure unbiased testing and to observe the behavior of the game in general.

Date	Millstone
4/07/08	Beta Release
4/11/08	general testing completed and bugs reported
4/14/08	bugs fixed
4/23/08	regression testing completed
4/25/08	Gold Release

The table below outlines our test cases and their outcomes.

TestID	Description	Input	Expected Output	Actual Output	Pass / Fail	Test Type	Comment
001	game loads correctly	run SpaceEx.bat	SpaceEx should be launched	SpaceEx is launched	pass	integration	
002	every button on main menu can be clicked	click on every button in main menu	button is clicked and action is performed	button is clicked and action is performed	pass	black box	
003	Every button on host menu can be clicked	click on every button in host menu	button is clicked and action is performed	button is clicked and action is performed	pass	black box	must use keyboard for map list - accepted
004	map corresponding to selection should be loaded	select each map in turn, click host	server should start with selected map	server should start with selected map	pass	integration	
005	correct credits screen is shown	from main menu, click Credits	Correct Credits screen should appear	Correct Credits screen appears	pass	integration	Must click escape after finished to go back to Main Menu – accepted
006	Tooltips on Main Menu are correct	Hover over items on Main Menu	Correct tooltips are shown	Correct tooltips are shown	pass	black box	
007	Tooltips on Host Menu are correct	Hover over items on Host Menu	Correct tooltips are shown	Correct tooltips are shown	pass	black box	No tooltips are shown for map list – accepted
008	Should be able to fire any SE weapon	From inside game, try to fire all SE weapons	All weapons fire correctly	All weapons fire correctly	pass	black box	
009	Switching weapons should display correct weapon name	From inside game, try to switch weapons	New weapon description show correctly	New weapon description show correctly	pass	black box	
010	Firing weapons should display correct	From inside game, try to fire all SE	Animations show correctly	Animations show correctly	pass	integration	



	animation	weapons					
<b>011</b>	Should not be able to fire weapons without ammo	Try to fire all SE weapons until out of ammo	Not able to fire weapons without ammo	Not able to fire weapons without ammo	pass	black box	
<b>012</b>	Switching weapons should switch corresponding HUD item	Try to switch weapons	HUD switch to correct item	HUD switch to correct item	pass	integration	
<b>013</b>	Correct ammo should be displayed in HUD	Try to fire all SE weapons until out of ammo	Correct ammo number displayed by HUD	Correct ammo number displayed by HUD	pass	black box	HUD displays ammo as ratio instead of as whole numbers - accepted
<b>014</b>	Killing bugs should increase your score	Try to kill many bugs	Increase bug score	Increase bug score	pass	black box	
<b>015</b>	Bugs should spawn	Look around map	Find bugs	Find bugs	pass	black box	
<b>016</b>	Can play network game	Setup server, try to join from other client	Clients can join and play on server	Clients can join and play on server	pass	integration	

# 12 Appendices

## 12.1 Project Team Description

### 12.1.1 Programming

- Nabil AlRamli (programmer)
- Brian Rodriguez (programmer)
- Jim Nardichia (programmer)
- Said Ketchman (programmer)

### 12.1.2 Art

- Adam Skutt (Character 3d modeler)
- Adam Reno (Level Designer)
- Adam Bishop (Prop/weapon/static modeler)
- Patrick Stannard (Project lead/Concept artist)

## 12.2 Asset List

### 12.2.1 Art

12.2.1.1 *Model and Texture List*

12.2.1.2 *Animation List*

12.2.1.3 *Effects List*

12.2.1.4 *Interface Art List*

12.2.1.5 *Cut scene List*

### 12.2.2 Sound

12.2.2.1 *Environmental Sounds*

12.2.2.2 *Weapon Sounds*

12.2.2.3 *Interface Sounds*

### 12.2.3 Music

12.2.3.1 *Ambient*

12.2.3.2 *“Action”*

12.2.3.3 *Victory*

12.2.3.4 *Defeat*

### 12.2.4 Voice

12.2.4.1 *Actor #1 lines*

12.2.4.2 *Actor #2 lines*

12.2.4.3 *Etc.*