

# Project Sirius

Design Document rev. 4

UCI CS113 “Team L”

© 2009 Reza Ghassemi, Jay Igarashi, Thomas Iu

Story by Reza Ghassemi

Technical Specification by Jay Igarashi

Gameplay Systems Design by Thomas Iu

## 1. Overview

The game is a 3d action-platformer with a behind-the-back point of view and controls similar to that of most modern first-person-shooters for personal computers. Other games within the same genre include the “Jak and Daxter” series and the “Tomb Raider” series. This game is built as a modification of “Unreal Tournament 2004” for Windows.

The player takes the role of Remmy the wolf as he escapes from Arcas city. Remmy is a wolfman in a futuristic world populated by anthropomorphic animal-people who have special powers and use advanced technology.

During each level of the game, the player controls Remmy with the primary objective of traveling to the level’s end. Completing each level furthers the story and brings Remmy closer to escape. To reach the end of each level, the player must traverse many platforming obstacles, such as pits, ledges, impeding walls, and elevated platforms using Remmy’s superhuman acrobatic abilities, such as the double-jump and air dash.

On some levels, the player must find and interact with particular non-player characters, items, or switches to accomplish mid-level objectives before being able to progress to the end of the level.

The levels are also filled with enemy animal-people and robots that chase and attack the player with special powers and hi-tech weapons. Remmy has his own attacks to use against the enemies, but none of his attacks can permanently defeat anyone. He can temporarily disable enemies with attacks such as the wolfwind, which knocks away any enemies within its area of effect, and the frostbolt, which is a projectile that freezes one struck enemy into an ice block for a few seconds. Since enemies cannot be eliminated, players must keep moving through the levels to evade enemy attacks. This is a game that emphasizes avoiding danger rather than fighting.

Remmy’s attacks are not only useful for stalling enemies, but are crucial to success in many platforming challenges. For example, even Remmy’s best jump may not be high enough to clear a particular wall, but he can freeze an enemy into an ice block, and then jump to stand on top of the block and gain some extra height from which to jump over the wall.

The game is suitable for young children, having a cartoon-like art style and low violence. Nobody ever gets killed or seriously injured. However, the intense platforming challenges and emphasis on speed and reflexes also appeal to older gamers who developed their reflexes during the days when the platformer genre dominated the game market. It is the type of game that works well for speed-running, a popular play style in which players try to traverse the levels of an action game as quickly as possible, and keep trying to improve upon their best times.

## 2. Game Specification

### 2.1 Mechanics

#### 2.1.1 Gameplay Controls

The player moves Remmy with the keyboard and rotates facing and aim by moving the mouse. The game assigns particular keys by default to control each action. Keys can be reassigned in the settings menu.

<b>Player Action</b>	<b>Key</b>
Move Forward, Climb Up	W
Move Left, Climb Left	A
Move Backward, Climb Down	S
Move Right, Climb Right	D
Jump, Double Jump, Wall Kickoff	Space
Dash, Air Dash	Double-Tap WASD
Toggle Crouch	C
Hang from Ledge	Hold R
Take Object, Interact with Environment, Release Ledge, Talk	R
Fire Ranged Attack	Left Mouse Button
Ranged Attack Select Menu	Hold Mousewheel
Next Ranged Attack	Mousewheel Down
Previous Ranged Attack	Mousewheel Up

##### 2.1.1.1 Move

Remmy runs in the chosen direction. Combinations of movement keys will cause Remmy to run in a diagonal direction, like forward and right. If facing an adjacent climbable surface, like a ladder, the forward key makes Remmy climb upward, and the backward key makes Remmy climb downward.

### ***2.1.1.2 Jump/Double Jump***

Remmy jumps his body height in altitude. Jumping may be combined with movement direction to determine direction of jump. Directional jumps travel horizontally at the same speed as Remmy's run. Once a jump has started, the player may not change its direction.

### ***2.1.1.3 Double Jump***

Remmy may also start a jump at any point in mid-air, gaining additional elevation as if the point in the air was a platform he can jump from. This ability may only be used once during each continuous period of time spent in the air. It may be used once during a jump or fall, but once used, Remmy must land before using it again. Using this ability enables Remmy to change movement direction in mid-air to the direction of the new jump.

### ***2.1.1.4 Dash***

Remmy can dash by double-tapping a movement key in the direction to dash toward. The dash is a half-second burst of very fast movement in the chosen direction, and is especially useful for dodging. Remmy does not run during this dash. Instead, he looks like he is flying just above the ground. Once a dash has begun, its direction may not be changed. After the dash is used, Remmy cannot dash again for another half-second. Initiating a jump during a dash cancels the remainder of the dash, and causes Remmy to jump in the direction of the dash.

### ***2.1.1.5 Air Dash***

The dash may be used in mid-air, which allows a half-second of high speed horizontal flight in the chosen movement direction. Using the air dash stops Remmy from falling during its duration. The air dash also cancels any upward momentum from jumping. After any air dash, Remmy starts falling as if from the apex of a jump. The air dash may be used only once during each continuous period of time spent in the air. It may be used once during a jump or fall, but once used, Remmy must land before using it again. Using this ability enables Remmy to change movement direction in mid-air to the direction of the air dash. If Remmy's normal dash takes him off a ledge, the dash continues as if it were an air dash, although a dash that began on the ground does not count toward the limit of one air dash usage per continuous period of time of air time. Initiating a jump during an air dash cancels the remainder of the air dash, and causes Remmy to jump in the direction of the dash. This air dash jump has the same speed and height of a normal directional jump and is the same as a double jump.

### ***2.1.1.6 Crouch***

The player presses the crouch button to make Remmy crouch or stand up from a crouch. Remmy can still move while crouched, but at half normal movement speed. Initiating a jump or dash cancels a crouch. Falling off a ledge does not cancel a crouch. Initiating a crouch while in mid-air causes Remmy to tuck his legs up beneath him, allowing him to clear higher obstacles and land on taller ledges. If Remmy crouches while in the air, he remains crouched when landing. The player may also toggle off the crouch before landing to untuck Remmy's legs and land standing. If Remmy is hanging from a ledge, crouching causes him to tuck his legs up as if he was in mid-air. Crouching halves the range at which enemies can initially detect Remmy. However, if a particular enemy has already detected him, crouching does not aid against detection by that enemy until Remmy first leaves the enemy's normal detection range or breaks the enemy's line of sight to him.

#### ***2.1.1.7 Roll***

Remmy may roll by initiating a crouch during a dash. Remmy rolls on the ground in the direction of the dash for up to 1 second at the same high speed as the dash. Rolling cancels the remaining duration of a dash. Toggling crouch off during a roll cancels rolling and causes Remmy to stand. If Remmy does not stand before the end of a roll, he ends the roll in a crouch. Initiating a jump or dash during a roll cancels the roll and makes Remmy stand. Rolls may not be performed in the air.

#### ***2.1.1.8 Hang from Ledge***

While adjacent to the top of a ledge, Remmy can grab onto it if the player is holding the grab ledge button. Grabbing a ledge is possible while in mid-air. After Remmy has grabbed the ledge, the player no longer needs to hold the grab ledge button to maintain the grab. Remmy holds onto the ledge with his hands and the rest of his body hangs down off it. While Remmy is grabbing a ledge, pressing the left or right movement buttons cause him to climb sideways along the ledge instead of Remmy's normal movement, while pressing the forward movement button makes him climb up on top of the ledge.

#### ***2.1.1.9 Release Ledge***

While grabbing a ledge, the player can press the release ledge button to have Remmy let go of the ledge and drop downward.

#### ***2.1.1.10 Take Object, Interact with Environment, Talk***

The same button that releases from a ledge grab also lets Remmy interact with an object, NPC, or environmental feature within touching distance of Remmy and aimed at with the

targeting reticule. If something in the game world is a valid target for interaction with this feature, a bracket appears around the target when the targeting reticule is pointed at it and Remmy is within range. Doors and switches are environmental features which may be targeted and used in this way. Important objects are picked up with this button as well. The game keeps track of the player's important objects. The player has no inventory management controls. Instead, each object is automatically expended or removed at some point in the game as appropriate to the story.

#### ***2.1.1.11 Fire Ranged Attack***

Remmy fires the currently selected ranged attack. Attacks may vary in size and the number of targets that can potentially be affected at once. Attacks originate from Remmy's body and are aimed toward the targeting reticule. Firing a ranged attack makes it unavailable for use, and causes its recharge timer to appear superimposed over the attack's icon. When the timer runs out, it disappears, and the attack becomes recharged and ready.

#### ***2.1.1.12 Wall Kickoff***

While grabbing a ledge at the top of a wall, or while in mid-air adjacent to a wall, Remmy can kick the wall to propel himself up and away from it. The wall kickoff makes Remmy travel the same vertical distance as a normal jump. This also moves him away from the wall in a direction perpendicular to the surface he is adjacent to, at the same horizontal speed as his run. As long as Remmy is in mid-air adjacent to a wall or grabbing a ledge, clicking or holding the wall kickoff button will activate the maneuver.

### **2.1.2 Physics**

The game uses Unreal's physics. There is gravity which resembles real Earth gravity. However, many characters can jump much higher than a real human, and falling is slower than real falling, which enables very long horizontal jumps. While running, characters can change direction instantly without being affected by momentum. Characters on the ground can instantly accelerate to their top speed, and also instantly stop. During a jump, horizontal direction and speed cannot be changed, however the double jump ability enables Remmy to start a second jump in mid-air, as if standing on a surface, and choose a different direction for the new jump. Attacks can exert force that knocks targets backward.

### **2.1.3 Shields**

#### ***2.1.3.1 Player Shields***

Remmy is protected by a force shield that reduces the potency of attacks that strike him so that he can quickly overcome their effects. However, each time he is struck by an attack, the shield's energy depletes by an amount depending on the attack. If he is struck while his shield is exhausted, the attack's effect incapacitates him. For example, if an enemy strikes Remmy with a frostbolt attack to freeze him immobile, and Remmy still has shield energy, the bolt only freezes him for a second and depletes some of his shield. If a frostbolt hits Remmy while he is without shield energy, he suffers long term incapacitation by freezing, and is automatically captured and defeated. If Remmy is incapacitated by an attack that hits him while his shield is drained, the player fails the level and must restart at a previous checkpoint. Remmy has a normal shield maximum energy of 100. Shields instantly recharge back to full strength if the user has not been struck by attacks for fifteen seconds.

### ***2.1.3.2 Enemy Shields***

Enemies are protected by force shields similar to Remmy's. Like Remmy, when they are struck by attacks while their shields are depleted, they are disabled to a greater extent than they would be by a shielded attack. Unlike Remmy, enemies cannot be captured and defeated, and eventually the effect of an unshielded attack wears off. For example, when Remmy strikes a shielded enemy with a frostbolt attack, the enemy is frozen for 5 seconds, during which it is unable to perform any actions. When Remmy strikes an unshielded enemy with wolfwind, the enemy is frozen for ten seconds. Different enemies have varying maximum shield energy.

## **2.1.4 Attacks**

### ***2.1.4.1 Player Attacks***

After the player uses one of Remmy's attacks, that attack cannot be used again until a particular amount of time passes. Each attack has its own recharge timer that determines the interval between uses. Since the same attack cannot be reused in rapid succession, players must think ahead about the best situation in which to use each attack.

The player starts with the Wolfwind attack and gains others after reaching specified places during the game levels. Attacks may not be lost.

When a shielded subject is struck by an attack, it suffers the attack's minor effect. When an unshielded subject is struck, it suffers the attack's major effect.

<b>Player Attack</b>	<b>Target</b>	<b>Shield Drain</b>	<b>Minor Effect</b>	<b>Major Effect</b>	<b>Recharge Timer</b>
Wolfwind	Very fast wave	10	Knocked back	As	5s

	strikes all enemies at close range in 60 degree arc in front of player		out to medium range and knocked prone; prone subjects spend 1 second standing up, during which they perform no other actions	minor effect and enemies cannot attack for 10 seconds	
Frostbolt	Slow, small projectile that flies in straight line; expended when it strikes a character or obstacle or reaches past medium range	15	Frozen into ice block for 5 seconds before reverting to normal; frozen subjects take no actions; characters may stand on block as temporary platform; subjects frozen in mid-air fall to the ground unless they have anti-gravity hovering, in which case they hover in place; targets ignore further shield drain and freeze effects while frozen	As minor effect but ice block lasts 15 seconds	10s
Icewall	Slow, small projectile that flies in straight line while user holds down attack button; stops moving and transforms into a vertical transparent wall when user releases attack button or switches	10 to any enemy struck by the projectile before it transforms into the wall or displaced by the wall as it forms; shield drain only affects	If the wall intersects any characters when the projectile first transforms into it, the characters are knocked away and displaced out of the wall into the closest space that the	As minor effect and enemies cannot attack for 10 seconds	15s

	<p>attacks, or when the projectile strikes a character or obstacle; if the projectile moves past medium range, it disappears without transforming into a wall; the height and width of the side of the wall facing the user are double the user's height; the wall is almost paper-thin; the wall is oriented so that one of its large sides faces the user; the wall is always perpendicular to the ground; the center of the wall is at where the projectile was when it stopped and transformed into the wall; the wall is immobile after creation; the wall may appear in mid-air and floats without regard to gravity; the wall functions as a normal wall in that it blocks character movement and projectiles, and can be used for a wall kickoff or ledge hang; the wall may intersect other objects and walls, but not characters;</p>	<p>enemies of the user</p>	<p>characters can fit in; characters may not stand on the wall, but instead slip off toward the closest side; the wall blocks movement and attacks from both the user and enemies</p>		
--	---	----------------------------	---	--	--



	the wall disappears 15 seconds after forming				
Vortex	Slow, small projectile that flies in straight line while user holds down attack button; stops moving and expands when user releases attack button or switches attacks, when the projectile strikes a character or obstacle, or when it reaches past medium range; when it expands, it instantly becomes a sphere of swirling wind with a radius equal to five times the user's height; the field disappears 15 seconds after expansion; only enemies in the field are affected, the field inflicts its effect and shield drain every quarter second to each enemy in the expanded field	1 per quarter second; user unaffected	Pulled toward center of field; distance over time equals the movement speed of medium speed enemies; user unaffected	As minor effect but enemies cannot attack while in the field	20s

### ***2.1.4.2 Enemy Attacks***

Like Remmy, enemies have attacks, with recharge timers that determine how often each attack can be used. Attacks deal some degree of shield drain, and have minor or major effects depending on the state of the target's shields. Some enemies have attacks that function similarly to Remmy's, however these attacks may work slightly differently from those the player has. Generally, enemy versions of the same attack have shorter recharge

times and shorter durations on their minor effects. Despite any differences on recharge times, shield drain, and durations, enemy attacks follow the same rules as player attacks with regard to allowing disabled targets to ignore further disabling effects and shield drain from attacks until half a second after any disabling effect expires. Enemy attacks can affect other struck enemies.

<b>Attack</b>	<b>Target</b>	<b>Shield Drain</b>	<b>Minor Effect</b>	<b>Major Effect</b>	<b>Recharge Timer</b>
Frostbolt	Slow, small projectile that flies in straight line; expended when it strikes a character or obstacle or reaches past medium range	15	Frozen into ice block for 1 second before reverting to normal; frozen subjects take no actions; characters may stand on block as temporary platform; subjects frozen in mid-air fall to the ground unless they have anti-gravity hovering, in which case they hover in place	As minor effect but ice block lasts 15 seconds; game over for player	2s
Crawlerbot Jump	Jump at high speed toward the target in a low arc; affects struck target and finishes jump arc to land; cannot act for 1 second after landing	15	Knocked back out to medium range and knocked prone; prone subjects spend 1 second standing up, during which they perform no other actions	As minor effect plus subject is stunned for 10 seconds; game over for player	3s

### 2.1.5 Player Attributes

Remmy's maximum shield energy and run speed are rated at 100 units, and represent the standard values against which all other shield and speed values in the game are compared. The other values can be considered percentages of these standard values.

<b>Remmy Attribute</b>	<b>Value</b>
Maximum Shield	100
Run Speed	100
Horizontal Jump Speed	100
Crouch Walk	50

Dash	175
Air Dash	175
Roll	175
Climb	50

## 2.1.6 Enemy Attributes

### 2.1.6.1 Pig Guard

Pig guards are anthropomorphic pigs wearing futuristic armored guard uniforms. Each wields a Frostbolt pistol.

The pig guard walks slowly in a patrol pattern until it gets line of sight to the player and the player is within its detection range, which is set to a long distance. If it detects the player, it approaches by running and attacks. The pig guard runs slightly slower than the player character.

The pig guard has a Frostbolt weapon that it keeps using to attack the player. The bolt travels slowly enough that it's easy to dodge at long range. The pig guard should try to enter medium range to increase its hit chance. It doesn't run away if the player gets too close.

The pig guard's attack should have a slow rate of fire. It should be able to fire once every 2 seconds. After firing, it visibly starts recharging, requiring 2 seconds to build up charge. If the weapon hasn't been fired recently because the pig guard hasn't engaged the player, the weapon should already be charged.

The pig guard should try to chase the player and enter medium range until the player has gone out of line of sight for 15 seconds. If the pig guard loses the player, it returns to its patrol pattern until it sees the player again. The pig guard also has a predefined zone which it never leaves.

It should always change its facing to point its weapon at the direction it is shooting toward. If not actively attacking, it points in the direction it travels toward.

<b>Pig Guard Attribute</b>	<b>Value</b>
Maximum Shield	60
Run Speed	80
Walk Speed	40

### 2.1.6.2 Hoverbot Security Bot

Hoverbots are compact robots about the size of a person's torso. They have no limbs, but each one does have an integrated frostbolt gun extending out from it. Hoverbots fly off the ground due to their anti-gravity engines.

The hoverbot normally flies slowly in a patrol pattern until it gets line of sight to the player and the player is within its detection range, which is set to a long distance. If it detects the player, it approaches and attacks.

It is able to change elevation during flight. Since its flight speed is slow, the player should be able to outrun it unless some difficult platforming is in the way.

The AI should keep firing its Frostbolt attack at the player. The bolt travels slowly enough that it's easy to dodge at long range. The hoverbot should try to enter medium range to increase its hit chance. It doesn't run away if the player gets too close.

The hoverbot's attack should have a slow rate of fire. It should be able to fire once every 2 seconds. After firing, it visibly starts recharging, requiring 2 seconds to build up charge. If the weapon hasn't been fired recently because the hoverbot hasn't engaged the player, the weapon should already be charged.

The hoverbot should try to chase the player and enter medium range until the player has gone out of line of sight for 15 seconds. If the hoverbot loses the player, it returns to its patrol pattern until it sees the player again. The hoverbot also has a predefined zone which it never leaves.

It should always rotate to point its weapon at the direction it is shooting toward. If not actively attacking, it points in the direction it travels toward.

The hoverbot can be frozen or knocked back by the PC's powers, but is immune to being knocked down.

<b>Hoverbot Attribute</b>	<b>Value</b>
Maximum Shield	10
Fly Speed	60

### ***2.1.6.3 Crawlerbot Security Bot***

Crawlerbots are small multi-legged robots that crawl on the ground. They stand just above knee height and have no arms or weapons.

The crawlerbot maintains a preset patrol pattern until it acquires line of sight to the player. If it sees the player, it moves toward the player until the player is within its jump range. It then faces the player and jumps forward to ram the player. The crawler has a

jump that travels fast and goes a significant horizontal distance before landing. The jump is faster than the player's run or jump.

After a jump, it can't move for 1 second, and can't jump again for 3 seconds, during which it crawls to stay within jump range. If the jump causes the crawler to hit the player, the player takes damage and gets knocked down.

When the crawler hits the player, it continues the trajectory of its jump while the player falls. If the crawler jump hits a wall, it drops back to the ground on its feet.

The player's attacks all affect the crawlerbot. If it gets knocked down, it stays stunned for 1 second before getting up. The knockdown causes it to fall flat on its belly. When it gets up, it uses its limbs to push itself back up. The crawler crawls fast, but slightly slower than the player runs.

The crawlerbot chases the player to stay within attack range. If the crawler loses line of sight to the player for 15 seconds, it goes back to its patrol routine until it sees the player again. The crawlerbot also has a predefined zone which it never leaves.

<b>Crawlerbot Attribute</b>	<b>Value</b>
Maximum Shield	10
Run Speed	80
Horizontal Jump Speed	200

## **2.1.7 NPC Conversation**

### ***2.1.7.1 NPC Comments***

Some NPCs can be spoken with by aiming at them with the targeting reticule and pressing an assigned button for interacting with objects, NPCs, and environmental features like doors and switches. Remmy and the NPC will then converse, with their dialog shown as text at the bottom of the screen. If an NPC is only meant to speak a one line comment when spoken to, the dialog takes place in real time during the normal action gameplay.

### ***2.1.7.2 Dialog Options***

In some cases, the player will be allowed to choose from multiple dialog options, which are numbered and listed in a column. If the conversation involves multiple lines or dialog choices for the player, normal gameplay is suspended, and the player enters a conversation game mode during which the normal action controls do not function, and the player instead uses the number keys or the mouse to select from the numbered dialog choices.

## **2.2 Artwork**

The game world and its characters are visually created with 3d models and animation.

Environments are futuristic science-fiction urban areas that mostly look metallic with abundant electronics and machines. There is a mix of indoor and outdoor locations. The game takes place in a hi-tech city with very tall buildings.

The art style overall is cartoon-like and unrealistic. Most characters are boldly colored and have larger heads and shorter legs than do realistically proportioned humans.

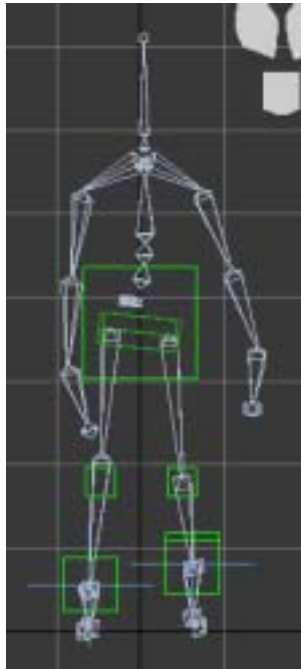
Menus, icons, and graphs displaying other game information are 2d graphics.

### **2.2.1 Characters**

#### ***2.2.1.1 Remmy the Wolf***

The player character (PC) is a wolfman wearing futuristic metal armor.





Remmy requires animations for his various actions.

Running forward	
Running backward	
Running left while body facing forward	
Running right while body facing forward	
Turning left while standing in place	
Turning right while standing in place	
Jumping up	
Jumping forward	

Jumping backward	
Jumping left	
Jumping right	
Double Jumping up	
Double Jumping forward	
Double Jumping backward	
Double Jumping left	
Double Jumping right	
Wall Kicking forward	
Wall Kicking backward	
Wall Kicking left	
Wall Kicking right	
Dashing	
Crouching from stand	
Stand from crouch	
Stand up from knockdown	
Crouch Walk forward	
Crouch Walk backward	
Crouch Walk left	
Crouch Walk right	
Rolling	
Hang from Ledge	
Pulling up from grabbed ledge to stand on ledge	
Shimmying on ledge left	
Shimmying on ledge right	
Attacking with Wolfwind breath while:	Standing still
	Running forward
	Running backward
	Running left while body facing forward
	Running right while body facing forward
	Jumping up
	Jumping forward
	Jumping backward
	Jumping left
	Jumping right
	Crouching
	Crouch walking forward
	Crouch walking backward
	Crouch walking left
	Crouch walking right
Attacking with ranged weapons in gauntlets while:	Standing in place
	Running forward



	Running backward
	Running left while body facing forward
	Running right while body facing forward
	Jumping up
	Jumping forward
	Jumping backward
	Jumping left
	Jumping right
	Crouching
	Crouch walking forward
	Crouch walking backward
	Crouch walking left
	Crouch walking right
	Jumping
	Running forward
	Running backward
	Running left while body facing forward
	Running right while body facing forward

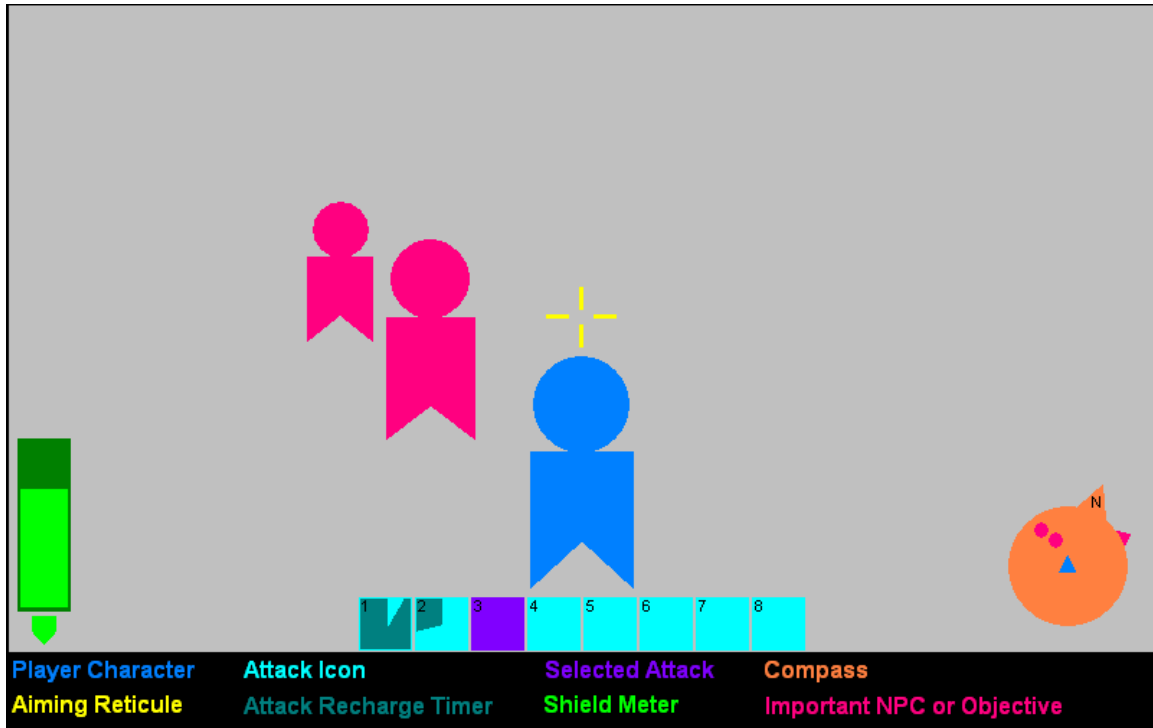
### ***2.2.1.2 Animal-People***

The other non-player characters (NPCs) in the game world are also anthropomorphic animals of various kinds, such as pig-people and sheep-people. Most of them will use futuristic tools and clothing like the player character.

### ***2.2.1.3 Robots***

The world has numerous NPC robots of widely varying designs, including flying guard robots and many-legged crawling robots.

## ***2.3 Graphical Interface***



### 2.3.1 Point of View

During normal gameplay, players see the PC from behind-the-back, and view the rest of the environment beyond the character. The PC always remains in the horizontal center of the lower half of the screen. A targeting reticule positioned at the center of the screen, slightly above the PC's head, shows where the player's attacks are aimed. When the player moves the mouse, the character and the reticule turn toward the direction of the mouse movement, and the point-of-view (POV) circles around the PC in an arc to stay behind the character and pointed at his back. The POV usually stays at the same distance from the PC's back: just far back enough to show the character's whole body in the lower half of the screen. The POV angle remains just high enough to look over the PC's head and see where the player is aiming, indicated by the reticule.

### 2.3.2 Compass

A circular compass in the lower right corner of the screen shows a bird's eye representation of the area around the player character. The player is shown as an arrow in the center of the circle, always oriented toward the top center of the compass. The compass has an arrow that travels its perimeter, always staying on the part of the perimeter corresponding to the north of the game world. The north arrow is labeled with "N" for north and always points north. Known subjects of importance, such as certain non-player characters or mission objective locations, appear as dots within the compass display if within close proximity to the player. If important subjects are beyond close

range, arrows at the perimeter of the compass point in the direction of the subjects, relative to the player's orientation.

### **2.3.3 Shield Meter**

A vertical bar graph in the lower left of the screen shows the player's remaining shield energy. Whenever shield energy is exhausted, the shield meter flashes and keeps flashing until it is no longer empty.

### **2.3.4 Attack Icon Bar**

Icons representing the player's various attack types are displayed at the bottom of the screen in a row.

#### ***2.3.4.1 Attack Recharge Timer***

Low opacity circular clocks superimposed over the attack icons indicate how much time remains before each attack type can be reused. The clocks represent remaining recharge time as a "pie slice." When an attack's timer expires, its clock disappears, and the attack icon flashes once to signal that it is once again ready for use. Trying to fire an attack that has not recharged does nothing except make a warning icon briefly blink in and out over the attack's icon.

#### ***2.3.4.2 Selected Attack***

The currently selected ranged attack type is highlighted to stand out from non-selected attacks. Rolling the mouse wheel one notch down highlights and selects the attack to the right of the currently selected ranged attack on the icon row. Rolling the mouse wheel up one notch selects the attack to the left of the current one. Rolling the wheel to select an attack beyond leftmost or rightmost icon will select the attack on the opposite end of the row from the current attack.

#### ***2.3.4.3 Attack Number***

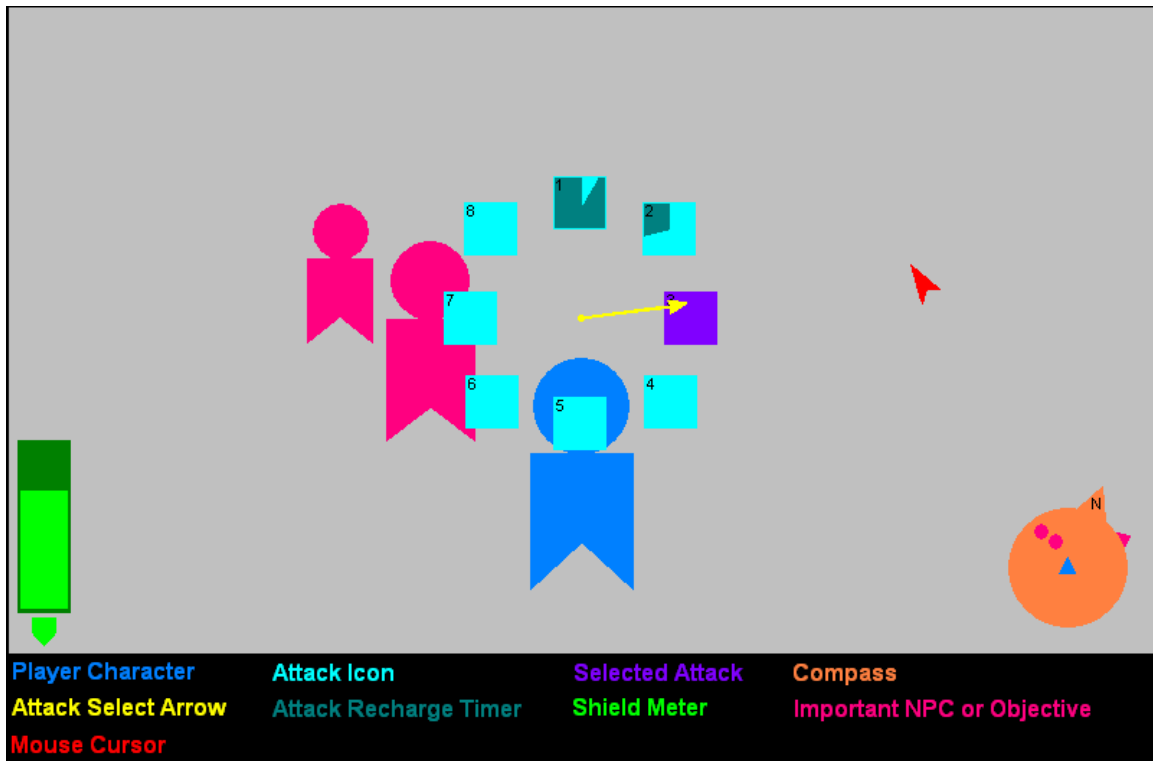
Each ranged attack icon is also labeled with a number. Pressing that number on the keyboard will select the attack. The attack icons are ordered with the lowest numbered attack on the left of the row, with the next lowest numbered attack adjacent to its right, and the each progressively higher numbered attack placed to the right of the previous number.

#### ***2.3.4.4 Firing Attacks***

Clicking the left mouse button fires the currently selected ranged attack. Most ranged attacks are projectiles that originate from the player character and travel in a direction indicated by the target reticule. After firing a ranged attack, that attack becomes unready and is deselected, while the closest ready attack to the right of the previously used attack becomes selected. If there are no ready attacks to the right of the previously used attack, then the closest ready attack to the right of the left end of the icon bar is selected. If there are no ready attacks at all, then the attack with the least time remaining before becoming ready is selected.

#### **2.3.5 Attack Selection Menu**

Holding down the middle mouse button activates the attack selection menu. When the attack selection menu is activated, it appears as a radial menu of icons in the center of the screen. The icons are evenly arranged in a circular formation. These icons also have superimposed circular clocks showing how much time is left before each attack is ready for use. The mouse cursor, unseen during normal gameplay, appears at the center of the attack selection menu when the menu is activated. The cursor disappears when the menu closes. An arrow originating in the center of the menu points toward the direction of the mouse. Whichever icon the arrow points to is highlighted to signify its being selected. The entire radial menu is superimposed over the action of the game world, and is low opacity, allowing the player to see what is happening through the menu. When the player releases the middle mouse button, the highlighted attack type is selected, and the attack selection menu and cursor disappear.



### 2.3.6 Title and Pause Menu

The pause and title menu lists options to continue, load, save, quit, change control settings, and start a new game. Menu items are listed in a column. In the menu, a mouse cursor appears to allow selection of menu items.

### 2.3.7 Save Menu

The save menu lists multiple save game slots in a column and the titles of the save files. Save slots can be selected by mouse cursor. The menu also has buttons to save or load the selected save slot or exit the save menu without saving or loading.

### 2.3.8 Control Settings Menu

The control settings menu has a mouse-click checkbox to toggle mouse-look inversion, and a slider to select mouse movement sensitivity. It also lists the in-game actions in a table column alongside another column listing labels of the keys assigned to activate those actions. By mouse-clicking the key label adjacent to a listed action, and then pressing a key on the keyboard, the player may assign the pressed key to activate the action in place of the previously assigned key. The new key assignment is saved and

displayed in the label. There are also clickable buttons to exit the settings menu or restore the default key assignments.

If a ranged attack selection key is reassigned to a different number key, on the gameplay screen, the attack is labeled with the new number on the row of attack icons. The icons are then rearranged so that the currently lowest numbered attack is on the left of the icon bar, and with the next lowest numbered attack adjacent to its right, and the each progressively higher numbered attack placed to the right of the previous number.

## **2.4 Audio**

### **2.4.1 Music**

The game has background music during its levels. The music is upbeat electronic dance music. The PC and important NPCs have leitmotifs during cutscenes also in an electronic music style. The music will be composed with electronic music software.

### **2.4.2 Sound Effects**

Various attacks like the frostbolt and the wolfwind require sound effects when they are fired as well as when they strike their targets. Sound effects will be acquired from free sources on the Internet.

## **2.5 Levels**

### **2.5.1 Level Descriptions**

#### ***2.5.1.1 Level 1, Tutorial***

The first level is a tutorial in which the player is taught to use Remmy's acrobatic abilities and special attacks. It is a simple platforming obstacle course with a few robot enemies. Textual hints appear to inform the player on what Remmy's abilities are and some of their possible uses. It takes place in the storage area of a prison after Remmy escapes his cell.

#### ***2.5.1.2 Level 2, Prison Courtyard and Observation Tower***

Remmy can move among the prisoners while avoiding the guards. The player meets many of the other characters in this part of the game. From speaking to the characters, Remy learns how to escape the prison through a weak point in the walls. However, he is soon discovered by the guards, and he jumps and climbs up the tall observation tower as he is chased. After he gets into the observation tower from the top, he can access the prison's teleporter, which can send him on a one way trip anywhere within the prison.

### ***2.5.1.3 Level 3, Detention Block***

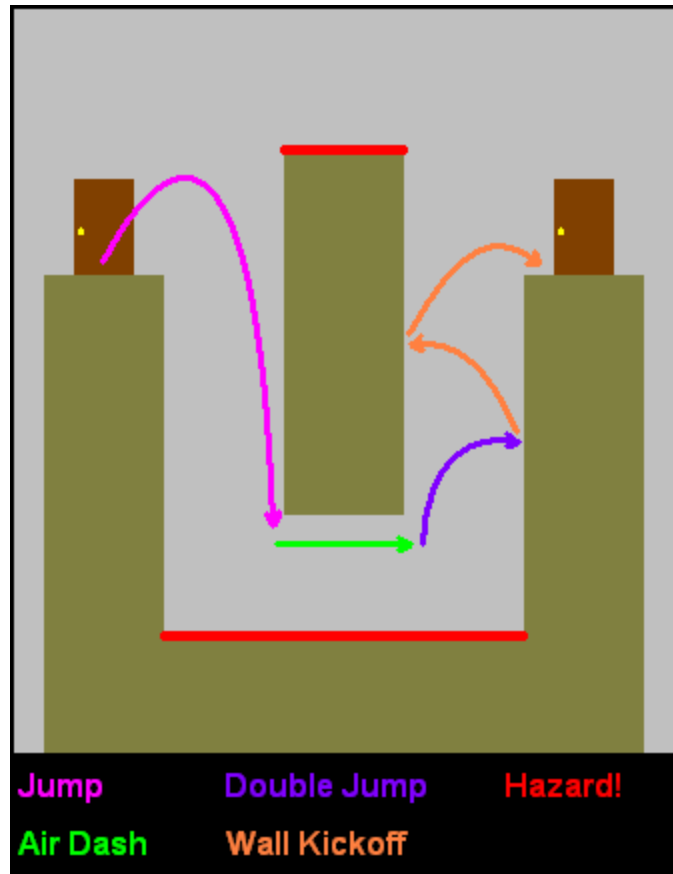
Remmy teleports from the observation tower to the detention block, where he frees one of the prisoners who helped him before. All the while, the player must avoid the guards and their robots. He then finds the weak point in the prison wall and breaks through it into the sewer.

### ***2.5.1.4 Level 4, Sewer***

The guards chase Remmy through the sewers. At the end, Remmy escapes into the city.

## **2.5.2 Platforming**

Levels focus on platforming challenges, which require the player to use Remmy's various jumping abilities and attacks to navigate. Easier challenges like in the tutorial level will simply require the player to use one special ability, such as using the wall kickoff to gain the extra necessary height to jump over an obstacle. Later on, advanced challenges will require stringing together of various abilities. The following diagram is a 2D side-view representation of how a player would use a series of special movement abilities to get across a room without touching any hazardous areas.



## 2.6 Story

### 2.6.1 Summary

Arcas city is corrupt city on Calisto. The inhabitants of Arcas city are all anthropomorphic animals of some sort. It has recently been struck with the (space) plague, and the incompetent police are trying to keep order by arresting anyone who might be sick.

An anthropomorphic wolf--whose name is REMMY, but we don't know that yet -- is unjustly arrested for coughing in public. He is thrown in the Arcas City Prison by unsympathetic GUARDS (anthropomorphic pigs). He shortly finds a way out by knocking out loose bricks in the floor, and escapes into the storage basement. The basement is a tutorial area. Remmy is an acrobatic character, and can also blow really hard with his Wolfwind attack (like the Big Bad Wolf).

After escaping the storage basement, Remmy finds himself in the Isolation Ward. There are a number of criminals here, but we can't see any of them because they are hidden behind thick sealed doors. They can, however, talk through text interfaces to someone on the other end of the door. There are no patrolling guards, because the isolation cells are



very secure (and because of the bad economy). The ward's entrance/exit only opens for those who the computer recognizes as belonging there. This includes the prisoners, since they are routinely taken to the exercise yards and such, so if Remmy frees one, they can help each other escape the prison. Problem is, some of the cell occupants are sick and Remmy doesn't want to get the plague. Computer logs indicate that only one of the prisoners is plague free, but not which one, and of course the prisoner's themselves will lie about it. This is a logic/interrogation puzzle, the player must identify the one healthy prisoner.

Remmy examines the doors locking mechanism (he makes his living as basically an interstellar tech help guy) and determines that if it gets really cold its failsafe will open the door and sound an alarm. Remmy can disable the alarm signal, but he doesn't have any way to generate cold. The prisoner tells Remmy that he can probably steal a high-end Frostbolt weapon from one of the guards post's in the interior of the isolation wards, so Remmy than has to go deeper into the ward, past security systems, in order to get it.

Remmy finds a guard asleep at his post (guarding the self-guarding isolation ward gets pretty boring), and steals his Frostbolt weapon from the weapon's locker. He installs it in his gauntlets. While he's at it, he paints a mustache on the guards face.

Remmy returns to free the prisoner by using the Frostbolt. The prisoner turns out to be a she-wolf person who introduces herself as JENNA (Balto reference). It is here that we learn Remmy's name, he introduces himself to Jenna. She won't say why she was arrested and in the Isolation Ward, but she has a sense of humor, so she and Remmy get along well. Not for long though, as right after they leave the ward the alarm goes off and they split up.

The guards don't come after Remmy so he assumes they must have just followed Jenna. It has activated SECURITY-BOTS, and Remmy must now deal with them while making his way toward the entrance. Before he reaches the entrance a Mysterious Voice appears on the intercom, apparently only in Remmy's current location. The voice tells Remmy that the guards have set up a trap near all the major exits for the escaping prisoners, and if he keeps going that way he will certainly be captured. Instead, he should head for the Exercise/Recreation Courtyard, as there's a prisoner there who might be able to help him. Since the voice apparently knows where Remmy is, and Remmy still isn't being captured, he figures it's safe to believe it and heads for the courtyard.

Remmy reaches the courtyard, where prisoners who don't have the plague are allowed to roam around under guard and security-bot supervision. There aren't many actual prisoners here though, since most of them either have, or are "under suspicion" of having, the plague. The two important characters in this area are a hacker named FLAPPER, and supervising guard, a huge wild boar-like anthropomorph nicknamed BRUTE. Brute is known as the toughest, and meanest, guard in the Arcas Prison. He moves around the courtyard, and will interrogate Remmy if he gets too close. On the third interrogation Remmy is found out, and its game over.

Remmy can talk to the few prisoners around here. The important one is Flapper, who knows of a structural weakness in the storage basement that leads into the sewer. Remmy can use his Frostbolt to break through it, which means its back to the beginning of the game. Flapper also mentions a woman asking questions about the prison's security network.

Before Remmy can leave, an alarm suddenly sounds. There is a commotion and it is evident that there is some ruckus in the Detention Block. Several guards arrive to "reinforce" the courtyard area (actually they are shirking their duty by escaping the danger zone). One of them helped escort Remmy to his cell in the beginning, and our hero is immediately recognized. Remmy quickly freezes Brute and runs away. The other guards are slow to follow because they fear Brute's wrath if they leave him. This isn't much help to Remmy however, because there is nowhere to go. He is backed up against the prison's Main Observation Tower, which doesn't open into the courtyard. Just in the nick of time, the mysterious intercom voice directs Remmy to a series of grooves in the side of the tower, which form the beginning of a way to climb it. With nowhere to go but up, Remmy complies.

Climbing the tower is intense platforming, with no guards or any other diversions -- just a difficult obstacle course. Finally Remmy arrives at an observation window which has been opened.

It's a surveillance room, with a wall full of monitors displaying different areas of the prison. The SURVEILLANCE OFFICER on duty is the source of the mysterious voice. His job is underpaid and unrewarding, and the prison won't hire another surveillance officer so he has to sit in this room all day, and most of the night, and watch nothing happen on the monitors. So, finally he got sick of it and started playing games with the guards and inmates for his own amusement. He saw Remmy draw the mustache on the Isolation Ward guard's face, and that made him laugh, a rare treat in his line of work, so he's been lending Remmy a helping hand. He won't reveal his name.

There is a Master Teleporter Room a few floors down the tower, and Mystery can set it to teleport Remmy to any other teleporter in the prison. Remmy wants to go to the teleporter in the storage basement, in order to escape.

However, Remmy also learns three other facts from Mystery: 1.) Brute is furious about being frozen by Remmy earlier, and has ordered that his capture is to be a top priority. All the guards will be on the lookout for him now, above anyone else. 2.) The ruckus that sounded the alarm earlier was caused by Jenna deactivating the security doors in the Detention Block. There are prisoners loose all over the place and guards trying to catch them. 3.) Jenna was caught, and is being held under guard in the Detention Block, until the riot is calmed down and she can be moved.

Remmy decides to save Jenna because he owes her one for leading off the guards after they escaped the Isolation Ward (and because he likes her). Mystery agrees to set the teleporter to send Remmy back to the Detention block. He can save Jenna and then run

back to his own cell, where the hole he made in the floor will lead back to the basement, from which he can break through the structural weakness and escape into the city sewers.

Remmy makes his way down the tower to the Teleport Room. Some of the guards the player has to deal with now have Frostbolt proof armor, since everyone now knows that Remmy has a Frostbolt . The teleporter works as intended, and sends Remmy to the Detention Block.

Now Remmy navigates through previously unexplored parts of the detention block. There are both guards and bots in this area, and ESCAPED PRISONERS, some of which have the plague and will run at Remmy. If he touches one of these prisoners, it's game over.

Remmy reaches an area of the detention block where they are holding Jenna. He freezes her guards, and sets her free. After he tells her the escape plan she quickly runs off. Remmy is disappointed she didn't wait for him, but doesn't have much time to think about it as Brute enters the room a few seconds later.

Brute and Remmy exchange insults, then Brute charges at Remmy. and this begins a long chase sequence. Remmy flees Brute through more of the detention area, dodging plagued prisoners on the way. He reaches the cell block where he was kept and jumps through the hole, only to have Brute follow him.

In the storage basement the player must navigate Remmy through the beginning tutorial at top speed, because Brute is right behind him. Remmy finds the entrance to the sewer (he had seen it the first time, just couldn't open it). It is already blown open. He runs inside.

It's a big sewer, with a river of sewage running the opposite direction. This is another intensive platforming level, this time with Brute pursuing. If Remmy makes a mistake, the sewer current will drag him back to the prison, and if he is too slow, Brute will catch him.

Finally Brute slips up and falls in the sewer, and Remmy is free. He reaches an exit out into the city. He climbs up. At the top, Jenna reaches down and helps him out, into the city.

## **2.6.2 Cutscenes**

Scripted non-interactive sequences will help tell the story of the game. These are in-game-engine cutscenes rather than separate movie files.

# **3. Technical Specs**

## ***3.1 Target System Specs***

Any system that runs Unreal Tournament 2004

Operating System:	Windows 98/Me/2000/XP
Processor:	Pentium III or AMD Athlon 1.0 GHz processor or faster (1.5 GHz or faster processor recommended)
Memory:	128 MB RAM minimum (256 MB recommended)
Hard Disk Space:	5.5 GB free
Video:	Any Windows-compatible video card(NVIDIA GeForce 2 or ATI Radeon with at least 64 megs of video memory recommended)
Sound:	Windows-compatible sound card. NVIDIA nForce(tm) or other motherboards/soundcards containing the Dolby Digital Interactive Content Encoder required for Dolby Digital audio
DirectX:	DirectX version 8.1(included)or higher
Multiplayer:	Internet (TCP/IP) and LAN (TCP/IP)play supported   33.6K baud modem or broadband Internet connection recommended

\*<http://www.unrealtournament2003.com/ut2004/specs.html>

### **3.2 Engine**

The Unreal Engine 2 will be used for this project in order to reduce trouble with 3D graphic processing, cutscene creation and level building. We will be using the Unreal Editor that comes with Unreal Tournament 2004 for building levels and modifying the existing scripts.

### **3.3 Programming Languages**

The Unreal Engine works with UnrealScript, a language created uniquely for Unreal. It comes with its compiler and tester which we will work with. Since there are licensing problems with directly editing existing code, new code will generally be created and have the game call to the new sub-classes.

### **3.4 Physics and Game System**

Most physics and game rules will be what exist in Unreal Tournament 2004. Those which will be changed are as follows:

Jump height:	Will be increased for more platforming capabilities
Wall kick:	Configure input associated to make it more user-friendly
Objective:	A level is completed when a particular area in the level is reached

New weapons:	Existing weapons will be replaced with those described in the Weapons section
Camera:	Will be made third-person view with more angle restrictions
Health bar:	Will be implemented under some other name such as Shield (needs revision)
Weapon select:	Will be picked from a radial menu upon right mouse click

### ***3.5 Input/Output***

Most will take the same as the default Unreal Tournament 2004. Such as:

WASD or arrow keys for movement
Mouse movement for camera angle
Mouse click for weapon fire
Space bar for jump/double jump

Some new input systems that will be implemented are:

New wall jump input
Middle mouse click for weapon select

### ***3.6 A.I.***

Enemies will patrol between navigation points and if the player enters their line of sight, they will chase after him if possible. Upon losing sight or moving too far away from the patrol area, they give up on chasing and return to patrolling. Core system will be using the AI from Unreal Tournament 2004 implemented through the Editor.

#### **3.6.1 Enemies**

##### ***3.6.1.1 Hoverbot***

The hoverbot floats off the ground and moves at slow speeds. It has a medium-range slow speed slow reload shot which reduces the Barrier of the player. Due to it floating, it can chase after the player to areas which other enemies cannot enter.

##### ***3.6.1.2 Crawlerbot***

The crawlerbot moves on the ground at medium speed. When chasing after the player, it also has a fast speed jump where it moves in a straight line attempting to tackle the player.

### **3.7 Weapons**

#### **3.7.1 Wolfwind**

The wolfwind is an area attack which pushes enemies away from you and possibly knocking them down on the floor. It is used to keep large groups of enemies from getting close. The player begins with this weapon.

#### **3.7.2 Frostbolt**

The frostbolt will make one enemy into an immobile ice block for a certain amount of time. The ice block can then be used as a platform to reach higher areas. The player receives this weapon after an event on the first level.

## **4. Personnel**

### **4.1 Ghassemi, Reza**

Responsible for Cutscene Scripting, Programming, Scriptwriting  
Needs to learn to program Unreal Engine with UnrealScript and what part of the existing code to modify to achieve desired results. Needs to learn to use UnrealEd to create cutscenes.

### **4.2 Igarashi, Jay**

Responsible for Programming  
Needs to learn to program Unreal Engine with UnrealScript and what part of the existing code to modify to achieve desired results. Needs to learn to import art and music and integrate it into Unreal Engine functions.

### **4.3 lu, Thomas**

Responsible for Level Design, Project Management  
Needs to learn to construct levels with UnrealEd.

#### **4.4 Ng, Trenton**

Responsible for Level Design, Music, Sound Effects  
Needs to learn to compose music with Ableton Live software.

#### **4.5 O'Neill, Ian**

Responsible for Art  
Needs to learn how to use 3D Studio Max to prepare model animations for use in Unreal Engine.

### **5. Deliverables**

#### **5.1 Seventh Week**

Ghassemi:	Write first level script; Program frostbolt attack, freezable characters, freeze special effects
Igarashi:	Program attack recharge timers, jump properties, double jump, air dash, wall kickoff, camera properties, trigger level win condition
Iu:	Design platforming challenges for level; prepare 3d props for use in level; import and organize art assets to make ready for use in Unreal Engine
Ng:	Construct functional first level with basic platforming
O'Neill:	Create basic 3d model of main character and make ready for animation

#### **5.2 Tenth Week**

Ghassemi:	Program wolfwind attack
Igarashi:	Program shields, HUD, game start/end conditions, enemy spawn properties; Create HUD art and shield special effects
Iu:	Expand first level to make use of new gameplay functions and enemies
Ng:	Compose first level background music
O'Neill:	Texture model; Improve polygon smoothness; Refine animations

#### **5.3 Final**

Ghassemi:	Program NPC dialog; Build cutscenes
Igarashi:	Organize code for distribution
Iu:	Finalize first level build
Ng:	Document gameplay for video
O'Neill:	Improve model texture and polygon arrangement