Initial Project Planning Sheet

# Brief Software Plan:

My game, Super Dragon Slayer Robot 9000 (working title), will be a top-down dungeon crawler, in which you start with minimal means of fighting oncoming monsters, and you must explore rooms to find weapons and power-ups. It will be visually similar to The Legend of Zelda (NES), with pixelated graphics. There will be various power ups which affect your offense, your defense, or your enemies directly. It will have typical WASD controls to move the character, and Arrow Key controls to use the weapon. Optionally, the player may control the character with an Xbox controller.

# Objectives:

# For each objective below supply a name, description, owner and a difficulty rating based on that owner

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Number** | **Objective Name** | **In-Software Description** (What is it, where will it be used in the software): 2 – 3 points | **Owner** | **Difficulty (1 to 10)** |
| 1 | Collision Detection | * The ability to detect when one in-game object touches or overlaps another, by using multiple forms of a collision-detection method * Will be seen when using weapons and projectiles | Ilan Segal | 4 |
| 2 | A.I. | * The program’s ability to independently decide how to control enemies * Path-finding * Aiming projectiles and weapons | Ilan Segal | 9 |
| 3 | Complex Sound System | * Sound effects and background music, dependent on in-game circumstances * Sounds coming from character and enemies * Music changing with things like location, or low health | Ilan Segal | 3 |
| 4 | Inventory System | * Giving the user the ability to manage their items, which affect the gameplay * Picking up items which have an immediate or passive effect on the game * Exchanging items for money (or vice versa) | Ilan Segal | 6 |
| 5 | Particle System | * Allowing the creation of “particle” objects, which behave according to conditions like gravity * Effects trailing behind objects as they move | Ilan Segal | 5 |