1. Introduction

1.1 RNG The game (This is not the final name of the game) is a game that says I am so used to RNG screwing me over in video games that I wanted to create a game that is fully at the mercy of RNG. Everything in this game is decided by random. How much damage you do, take, heal, coins you receive, even items you buy! In short it will be a fun RPG that will likely never play the same because of the absurd amount of RNG!

1.1 Purpose

1.1.1 The purpose of the game is to increase my programming portfolio, possibly make money, and most importantly, increase my capabilities as a programmer.

1.2 Intended Audience

1.2.1 The intended audience are people who like RPG’s, people who like video games, streamers who would love to play this game on stream for the pure chaos of the experience. And for anyone who would enjoy the pure chaos of playing a game where you have very little control of how your own choices play out as well as the rest of the game's choices as well. It could also be intended for those who play video games for the story rather than the game itself.

1.3 Intended Use

1.3.1 For fun! Or to prove a point? Or to discuss the nature of RNG itself. Most of all the intended use is to be played and to enjoy the experience.

1.4 Product Scope

1.4.1 Development Timeline

1.4.1.1 What would I develop it on?

1.4.1.1.1 Visual Studio Code

1.4.1.1.2 Godot

1.4.1.1.3 Unreal

1.4.1.2 What programming language should I use to develop it?

1.4.1.2.1 C++

1.4.1.2.2 C#

1.4.1.2.3 Python

1.4.1.2 weekly effort allocation

1.4.1.2.1 goal per week

1.4.1.2.1.1 Once all of the documentation has been taken care of I plan on working 5 hours a week during work and the other 4 during my personal time.

1.4.1.3 full development timeline

1.4.1.3.1 \* development timeline may have to be adjusted due to issues arising with getting the framework of the ground… (1/20/25)

1.4.1.3.1 Month 1

1.4.1.3.1.1 the full skeleton built and much of the base mechanics fleshed out/working towards available testing by month two. Have the map able to be explored

1.4.1.3.2 Month 2

1.4.1.3.2.1 Finish the basic moments. Develop and be able to start testing the combat system. Start work in the menu system.

1.4.1.3.3 Month 3

1.4.1.3.3.1 manage and develop the maps, test out and balance the boss battles. Work on implementing the unique character abilities and armor

1.4.1.3.4 Month 4

1.4.1.3.4.1 work on leveling up adjustments. The assignment of different rarities of weapons and armor. Test RNG to see if it is still as random as I thought it should be.

1.4.1.3.5 Month 5

1.4.1.3.5.1 Testing is the main focus of this month and should only be making changes based on play testing. Focus on refining the difficulty slider and effective RNG randomness where possible.

1.4.1.3.6 Month 6

1.4.1.3.6.1 clean up programming. Making things look pretty, functional and effective. Rigorous play testing will be needed by the time I am ready to make this game available for play.

1.4.2 Game storage size

1.4.2.1 I want the game to be smaller in size (storage wise) but I still want a full fledged game.

1.5 Definitions and Acronyms

1.5.1 RNG

1.5.1.1 Random Number Generator

1.5.2 RPG/RPG’s

1.5.2.1 Role-Playing Game

1.5.3 explore\*

1.5.3.1 This game will likely sit somewhere between a linear pathing (like pokemon before gen 9) and a completely open world (like Zelda TOtK). I don’t know exactly what that will entail at this moment in time but I want to make sure that my players are not on railroad tracks and they have the ability to explore the world but it won’t be completely open world. I may have to do a combination of the spin-off Mario games (Super Mario RPG for example) and (insert game here… I can’t think of one at this moment) 8/20/24

1.5.4 lvl

1.5.4.1 an abbreviation for level

2. Overall Description

2.1 This RPG is a fun adventure game that is tormented by RNG! In this world an evil wizard has cursed the land to the terrible power of inconsistency! You are tasked with finding this wizard's lair and getting him to put things back as they used to be. You are to traverse through the world, fighting enemies, recruiting teammates, obtaining weapons, upgrading weapons, armor, and abilities to gain enough strength to defeat the terrible wizard. This turn-based combat system gives you the freedom to order the lineup of your team to best combat the opponents you are facing. There is some strategy involved but you must truly prove your merit by how well you handle the constant onslaught of RNG in addition to formulating a strategy that doesn’t end in your doom. This game is not fair, it is random!

2.1 User Needs

2.1.1 replayability factor

2.1.1.1 how the RNG “changes” things

2.1.1.1.1 Range increase on the top end

2.1.1.1.1.1 When you will level up a weapon, armor, or ability you can increase the top end of the range so that you keep all the previous possibilities but you add a few more higher options to increase the possible damage output. The RNG chances to get any number in the range then adjusts to give each number possibly the same chance of being rolled. One level up range is 1-5. What number you get is also random. Example: sword lvl 1: damage 1-10. Sword lvl 2: damage 1-13. These will affect weapons armor and abilities.

2.1.1.1.2 Range decrease on the bottom end

2.1.1.1.2.1 If you level up an item you can make the range of possible value more consistent by upgrading the bottom end. With each level increase you can increase the bottom, thereby decreasing the RNG total range. That range increase would be from 1-3 points. Example: sword lvl 1: damage 1-10. Sword lvl 2: damage 2-10. These will affect weapons armor and abilities.

2.1.1.1.3 Damage increase index

2.1.1.1.3.1 If you level up an item this way instead of affecting the RNG all you affect is the overall range of your damages by 1-2 points. Example: sword lvl 1: damage 1-10. Sword lvl 2: damage 3-12. These will affect weapons armor and abilities.

2.1.1.1.4 loot/item spawns/gifts

2.1.1.1.4.1 The kinds of weapons the enemies drop upon defeat, items found while exploring, and gifted to the player will all be randomly selected from a loot table. This loot table will likely change up and or chances of certain items will be more or less likely based on the difficulty the player chose at the start of the game.

2.1.1.1.4.2 The loot table will also apply to in-game shops and each seller will display random items for sale and give a random amount of money needed to purchase said item. The range for in shop items will always be the same but how much is the base number for what an item will cost is dependent on several things.

2.1.1.1.4.2.1 weapon/armor rarity(common, uncommon, rare, legendary)

2.1.1.1.4.2.2 weapon type/armor type (Bow, ax, sword, etc./made of leather, iron, etc.)

2.1.1.1.4.2.3 weapon/armor level

2.1.1.1.4.2.4 for the loot table it will need to be generated based on a few things:

2.1.1.1.4.2.4.1 level (100)

2.1.1.1.4.2.4.2 rarity (5)

2.1.1.1.4.2.4.3 armor type (4)

2.1.1.1.4.2.4.4 weapon options (9?)

2.1.1.1.4.2.4.5 healing items (5?)

2.1.1.1.4.2.4.6 some parts of the loot table would only be accessed for shops/certain specific times that drops are guaranteed

2.1.1.1.4.2.4.7 gold (100?)

2.1.1.1.4.2.4.8 nothing (10% chance)

2.1.1.1.5 enemies per encounter

2.1.1.1.5.1 This will be a two fold RNG issue. It will partly be based on the difficulty of the player. It will also be affected by the current power level/strength of the player. I want the difficulty to scale such that as you get stronger the enemies are accordingly made stronger so that there will feel like a sense of progression is made because of the increased difficulty.

2.1.1.1.6 accuracy of attacks

2.1.1.1.6.1 Each close range weapon for the player will have 95% accuracy unless the weapon shows otherwise. Each long range weapon will have a 75% accuracy unless shown otherwise.

2.1.1.1.6.2 The enemies accuracy will be much more variable. I will have a base percentage based on the difficulty chosen. Including a base bonus for what kind of enemy it is. (bow user, sword user etc) Then I will have a variable percentage added based on the strength/power level of the player. These percentages are normal difficulty they have 50% accuracy, hard they have 65% accuracy, Difficult they have 80% accuracy, and unfair they have 99% accuracy

2.1.1.1.7 Enemy abilities

2.1.1.1.7.1 Depending on the enemies you are facing and what difficulty you are playing on, and the player's power level. Enemies abilities will be more or less potent.

2.1.1.2 Various weapons

2.1.1.2.1 Weapon Rarity

2.1.1.2.1.1 Common rarity

2.1.1.2.1.1.1 A common weapon will have three things randomized. The basement, the ceiling, and the accuracy. Example: sword lvl 1: damage (1-4)-(10-18) max level 30

2.1.1.2.1.2 Uncommon rarity

2.1.1.2.1.2.1 An uncommon rarity weapon will maintain the same kind of randomizations but it will have a guaranteed boost in stats compared to the common weapon. Example: UC sword lvl 1: damage (3-7)-(12-25) max level 50

2.1.1.2.1.3 Rare rarity

2.1.1.2.1.3.1 A rare weapon will be stronger than an uncommon by two boosts compared to the uncommon weapon. Example R sword lvl 1: (7-15)-(16-39) max level 70

2.1.1.2.1.4 Legendary rarity

2.1.1.2.1.4.1 The legendary weapon is guaranteed a \*unique ability but its damage is three boosts above the rare weapon. Example L sword lvl 1: (13-30)-(22-61) max level 100

2.1.1.2.2 close range weapons

2.1.1.2.2.1 close range weapons will have greater accuracy than long range weapons in general. Close range will be more effective against certain enemy types that long range weapons struggle with.

2.1.1.2.3 long range weapons

2.1.1.2.3.1 Long range weapons will be less accurate overall but able to hit enemies that are farther away. They will also be more effective on certain enemy types as well.

2.1.1.2.4 unique weapons

2.1.1.2.4.1 Unique weapons may not be bound by the same rules as the other two types and the uniqueness. There won't be as many restrictions on them because of the quirkiness of their uniqueness.

2.1.1.3 Various armors

2.1.1.3.1 armor rarity

2.1.1.3.1.1 There are four types of armor. Common, uncommon, rare, and legendary.

2.1.1.3.1.1.1 common armor reduces damage taken anywhere from 5% to 15 % per hit. The base chance of this occurring is 20%

2.1.1.3.1.1.2 uncommon armor reduces the damage taken anywhere from 10% to 20% per hit. The base chance of this occurring is 25%

2.1.1.3.1.1.3 rare armor reduces the damage taken anywhere from 15% to 25% per hit. The base chance of this occurring is 30%

2.1.1.3.1.1.4 legendary armor reduces the damage taken anywhere from 20% to 30% per hit. The base chance of this occurring is 35%.

2.1.1.3.2 armor make

2.1.1.3.2.1 Wood (weakest)

2.1.1.3.2.2 Leather

2.1.1.3.2.3 Iron

2.1.1.3.2.4 Steel (strongest)

2.1.1.3.3 enchanted armor

2.1.1.3.3.1 Enchanted armor can have three different abilities to be decided at random when generated.

2.1.1.3.3.1.1 A 10% increase of effects on the armor to trigger.

2.1.1.3.3.1.2 A 15% decrease of damage when the chance triggers

2.1.1.3.3.1.3 A 5% chance to take no damage

2.1.1.3.4 customizable skins

2.1.1.3.4.1 each armor and armor piece will be customizable. You can change the color,style, shape, and look.

2.1.1.4 Various character abilities

2.1.1.4.1 all character abilities need to be scalable. I also need to make sure that whatever I choose is scalable to what amount of levels… Max level will be 100. 8/27/24

2.1.1.4.1.1 level 1: All weapons gain a 1% chance to critical hit. Max level: all weapons gain a 50% chance to critical hit.

2.1.1.4.1.2 level 1: player takes 1% less damage on all hits. Level 100: player takes 30% less damage on all hits

2.1.1.4.1.3 Level 1: the player gains 5% critical hit chance on all short range weapons. Level 100: the player gains a 65% chance to critical hit on all short range weapons

2.1.1.4.1.4 Level 1: the player gains 3% accuracy increase on all long range weapons. Level 100: the player gains a 40% increase to accuracy on all long range weapons.

2.1.1.4.1.5 Level 1: the player gains a 5% increase to all money received at the end of battles. Level 100: The player gains an 80% increase to money received in all battles.

2.1.1.4.1.6 These are just a few examples. I will delve more into it when actually making the game fully to have a better idea of what I could incorporate.

2.1.1.5 Ability to explore\* many places

2.1.1.5.1 Like pokemon before gen 8 the maps will have many places to travel and many different ways to get there. It won’t be a full open world experience but you will have quite the range of freedom to traverse the different paths that exist. I intend to make it so that each city will be the safe haven between the dangerous roads that you must travel down. I am against fast travel methods… as of now 8/27/24.

2.1.1.6 Difficulty sliders

2.1.1.6.1 in order to maintain a sense of difficulty on each and every play through. They can choose a mode. Their options are normal, hard, difficult, and unfair.

2.1.1.6.1.1 These difficulty sliders will affect many parts of the game. The biggest thing it will affect is the power and strength of all opponents. This will be a basement increase but it will scale with the players level and the amount of progression of the player. It will also increase the overall health of opponents so that it is hard to kill them.

2.1.1.6.1.2 hard, difficult, and unfair will be locked until you have beaten all the easier game modes.

2.1.1.6.2 as the player progresses the normal enemies will scale based upon the progression of the player. (usually it will be based on how many bosses they have beaten) The more bosses that have been defeated the better item rarity they will have in addition to whether or not the armors or weapons are enchanted are affected by this as well. In addition the better armor they will have.

2.1.2 Simple controls

2.1.2.1 method of controlling the game

2.1.2.1.1 console controllers (xbox, PS5, etc)

2.1.2.1.1.1 moving of the character left control stick

2.1.2.1.1.2 menu navigation (up and down) left control stick

2.1.2.1.1.3 selecting highlighted item (X or A button)

2.1.2.1.1.4 deselecting/backing out of that part of a menu (circle or B button)

2.1.2.1.1.5 weapon toggle (triangle or Y button)

2.1.2.1.1.6 bringing up the menu/closing the menu out completely (square or X button)

2.1.2.1.1.7 pausing the game (start button)

2.1.2.1.1.8 toggling gameplay speed (increase speed - right on the d-pad. Decrease speed - left on the d-pad)

2.1.2.1.1.9 toggling information displays (basically it will have three different modes of information it will display in any menu/window they are in.) Display options are simple, informative, advanced.

2.1.2.1.1.9.1 left trigger to toggle between the different options. Right trigger to lock/unlock the display option chosen.

2.1.2.1.2 desktop setup

2.1.2.1.2.1 moving of the character (W, A, S, D)

2.1.2.1.2.2 menu navigation (up and down) scroll wheel/scroll bar

2.1.2.1.2.3 selecting highlighted item (mouse click)

2.1.2.1.2.4 deselecting/backing out of that part of a menu (clicking the x button/can be programed)

2.1.2.1.2.5 weapon toggle (T key)

2.1.2.1.2.6 bringing up the menu/closing the menu out completely (E key)

2.1.2.1.2.7 pausing the game (P key)

2.1.2.1.2.8 toggling gameplay speed (increase speed - R key. Decrease speed - F key)

2.1.2.1.2.9 toggling between the different informational display options

2.1.2.1.2.9.1 (1 key for simple, 2 key for informative, 3 key for advanced. 4 key to lock/unlock the display options

2.1.2.1.3 alternate controllers (button/control remapping to any device for accessibility)

2.1.2.1.3.1 I will provide a menu/area in the controls that will let anyone rebind the keys however they want.

2.1.2.1.4 stretch challenge make it available on mobile? I won’t define how the controls work for now just because I don’t know if I want this one mobile.

2.1.3 Simple UI

2.1.3.1 when the player first starts the game all of the settings will default to the simplest and base modes. There won’t be any harder difficulties available until they have cleared the game once. There will also be options to toggle how much information is shown on each window or portion of the game. (Mostly because a little kid may only want the simple layout when an adult would want all the information that they can get.) There is also something to consider when it comes to the menu that adjusts what the player is holding/wearing versus what is in the inventory.

2.1.4 customizable settings

2.1.4.1 There will be several things that are customizable. Here is just a list of some of the things that the player can customize.

2.1.4.1.1 menu borders

2.1.4.1.2 text speed

2.1.4.1.3 speedrun mode

2.1.4.1.4 damage numbers being displayed on attacks

2.1.4.1.5 information box pop ups (when you hover the mouse over it but not selected it yet.)

2.1.4.1.6 health bars

2.1.5 Platform availability

2.1.5.1 This is just a reminder that the game will be eventually made for consoles as well as PC so that anyone can play it. Stretch goals port it for switch as well 9/9/24

2.2 Assumptions and Dependencies

2.2.1 Game beatability

2.2.1.1 I do want to make sure that the game is beatable by most people, young and old. So I will need to think of how I will adjust the difficulty/make the game easier or harder based on how the player has progressed. It will likely take a lot of tweaking behind the scenes and playtesting quite a bit to make sure the game has a balance of difficulty and beatability. (I could add luck boosters…. To make the game beatable for certain parts of the game)

2.2.1.2 Experience scaling. The longer the game goes on the more experience you can gain to level up your character. I will need to also figure out how to scale this so that each of the battles has a reason to fight it but it becomes less needed as you have put in more effort.

2.2.2 RNG, the double edged sword

2.2.2.1 Since RNG is baked into every aspect of the game the eventual marketing of this game must make it clear that I only did so much to make it reliable. The vast majority of the game is at the mercy or blessing that is RNG. I still think the tagline of this game is This game is not fair, It’s random. Since a ton of the game is at the mercy of RNG does the inherent randomness make the game fair and balanced?

2.2.3 Balanced gameplay

2.2.3.1 One of the easy ways balance can be often obtained is there will be situations that will better determine the balance. If the balance of strength in numbers versus strength in the few. This will be noticed mostly in boss battles, and random enemy encounters.

2.2.4 the scope of “open world”

2.2.4.1 The true scope of this world is to somewhat limit the player on how/where they go. I only want to do that by not making a true open world but an open path world. There will be cities they can go to and different roads they can take to get to them. Each path will have its ups and downs and some paths may only be accessible one direction. Others may be more dangerous if you are a lower level but you won’t be barred from there if you don't have a high enough level.

2.2.5 effective/informative tutorial

2.2.5.1 tutorial option.

2.2.5.1.1 I will have it that the player will have the opportunity to go do the tutorial at any point in the game. So if they chose to skip it entirely then that is up to them. I will give experience/levels to the player if they complete it so that is the incentive to play the tutorial.

3. System Features and Requirements

3.1 Functional Requirements

3.1.1 that it is able to run on “modern” consoles and most PC’s that have been around in the last 10 years

3.1.2 that the graphics are simplistic enough in design in order to not drain the computer too much.

3.1.3 forgiving responding to players. This game will not require any quick time/fast reactions to the events going on. It will be more of how well you strategize against your opponents that will matter more.

3.1.4 effective health scaling for the player and the enemies

3.2 External Interface Requirements

3.2.1 The interface will always require either a keyboard and mouse or controller. Touch screen will only be made available if I decide to make a mobile port.

3.3 System Features

3.3.1

3.4 Nonfunctional Requirements

3.4.1

Random ideas to possibly implement

Random chance to crash the game

Random price point for purchasing the game

Using several randomized algorithms, not just one.