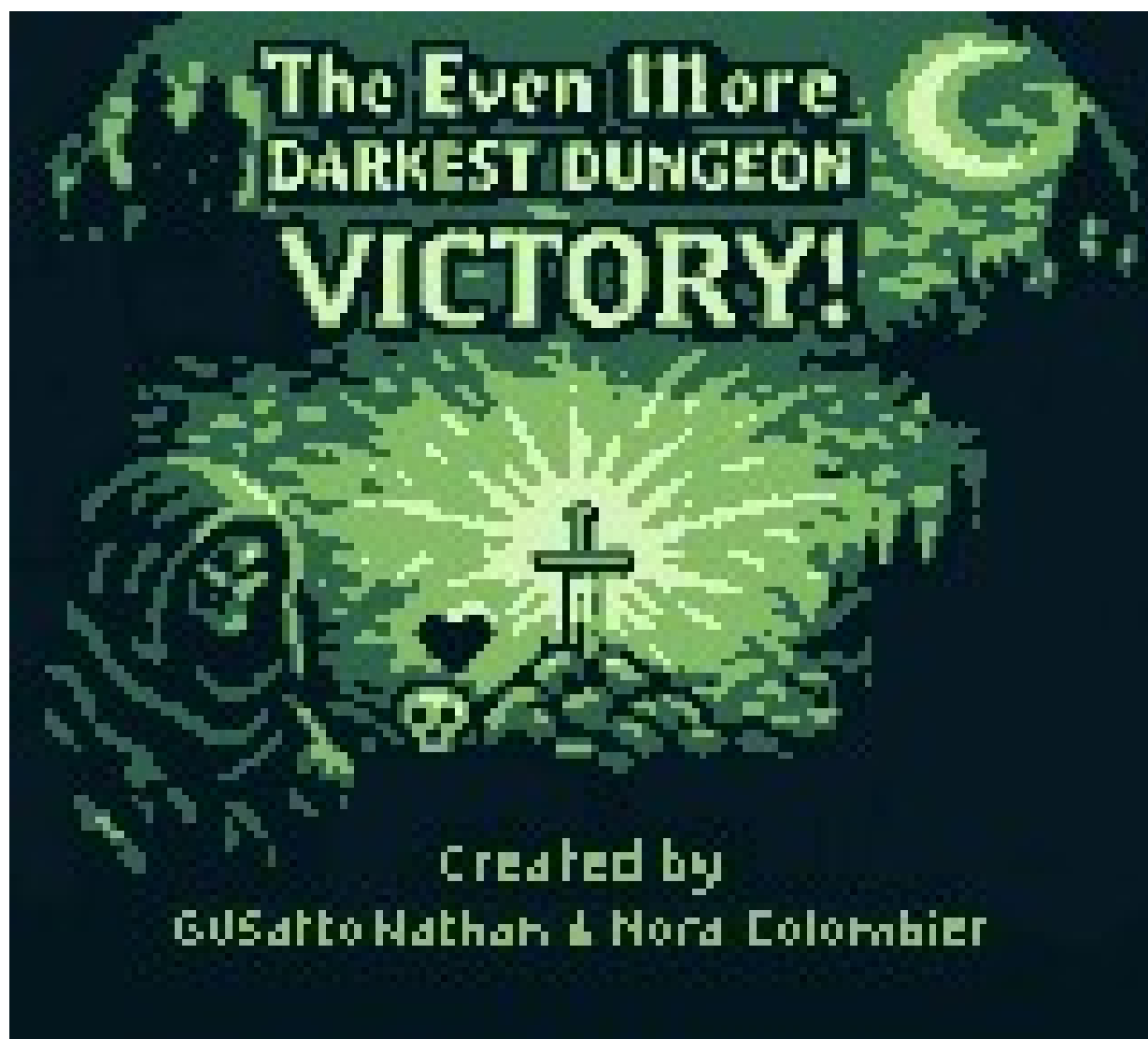


# The Even More Darkest Dungeon

Gusna453 – INDRO2 – 2025-2026



# Table of Contents

Introduction .....	4
Concept.....	4
Specifications / Constraints .....	4
Hud .....	4
Health Bar .....	6
Enemy Health Bar .....	6
Fighter Skill Level .....	6
Healer Skill Level.....	6
Controls .....	7
Move .....	7
Interaction.....	7
Combat.....	8
Combat Loop .....	8
Experience And Leveling .....	9
Levels / Screens.....	10
Assets .....	10
Scenes .....	10
Tilemap .....	11
Sprites .....	11
Sounds / Music .....	12
Game Flow.....	13
Main Flow .....	13
Level Flow .....	14
Level 1.....	15
Level 2.....	15
Level 3.....	16
Level 4.....	16
Last Scene.....	17

Critical Path / Design .....	18
Appendix .....	20
Glossary .....	20
Tools / Softwares .....	20
Credits.....	20

# Introduction

## Concept

The Even More Dungeon is a 2D TopDown Dungeon Crawler. In this game, the player incarnates a knight that enters a dungeon to find and get the legendary lost sword: Excalibur.

The player needs to explore floors and defeat monsters to level up his skills in fighting and healing to defeat the ultimate boss, which is keeping the sword, without dying from enemies.

The game has been inspired by games like Darkest Dungeon or Dungeon Masters (Atari).

## Specifications / Constraints

The game has been made for GameBoy Classic which means:

- GB Studio as Game Engine.
- A Screen Resolution of 160x144px.
- A color palette of 5 colors:
  - o #65ff00
  - o #071821
  - o #86c06c
  - o #e0f8cf
  - o #306850
- Sounds: 8 bits with format .mod (for game engine compatibility).
- Graphisms artstyle: Pixelart (of course).
- 4 Directions, A, B, Start and Select buttons.
- <https://www.nintendo.com/en-gb/Support/Game-Boy-Pocket-Color/Product-information/Technical-data/Technical-data-619585.html>

## Hud

The following picture shows the player in-game displayed HUD.

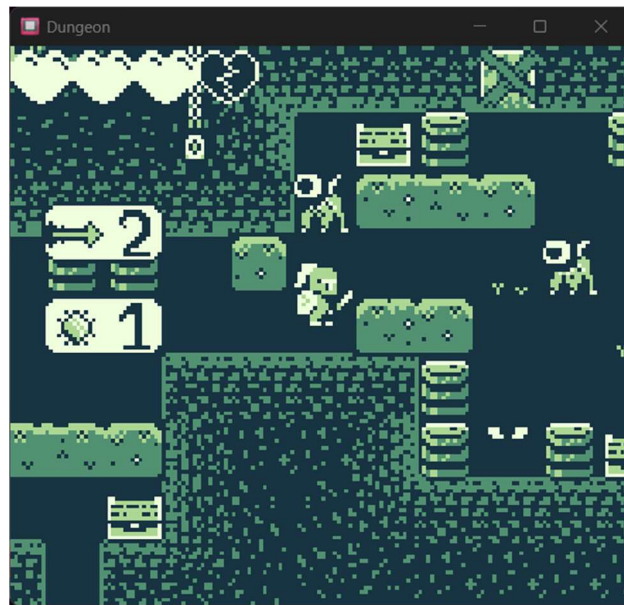


Figure 1 - Not in combat



Figure 2 - In combat

## Health Bar

The health bar displays the player's health at the very top left corner. The player can have up to 8 HPs<sup>1</sup> and each state is displayed by a combination of full / half-full / empty hearts (respectively 2, 1 or 0 HPs).

The health bar is updated at the end of each combat turn, after the enemy has potentially attack (cf. [Combat Loop](#)).

## Enemy Health Bar

The enemy health bar (exactly as the player's health bar) displays the enemy current health right below the player's health bar.

This part of the HUD only appears when the player enters combat and is hidden when the combat is ended (the player or the enemy dies).

## Fighter Skill Level

The Fighter skill level is displayed at the left side of the screen and is represented by an arrow with a number displayed on his side (the current fighter skill level: cf. [Experience And Leveling](#)).

## Healer Skill Level

The Healer skill level is displayed at the left side of the screen (right under the fighter skill level) and is represented with a number displayed on his side (the current healer kill level).

---

<sup>1</sup> Health Points

## Controls

### Move

The player can move with the D-Pad<sup>2</sup>. The player sprites have 2 frames for each direction the player is moving to (except the side which is simply inverted from the other side animation: cf. [Sprites](#)).

### Interaction

The player can interact with actors by penetrating their hitboxes:

- Chests: Heal the player for 1HP.
- Key Chest: Give the player the key to access the last level and therefore win the game.
- Enemies (cf. [Combat](#))
- Doors: Open the door.

In some cases, for example chests, further dialogue could be shown. They can be handled with the press of A button.

---

<sup>2</sup> Directional Pad

# Combat

## Combat Loop

A combat phase starts if the player enters an enemy hitbox.

The player cannot flee an enemy.

A combat phase only stops whenever the player or the enemy dies.

First, an enemy spawn with a random number of HPs (capped at 8) which is calculated as follows:

$$\$EnemyHP = (\$HealerLvl + \$FighterLvl + \$WizardLvl) + rnd((\$HealerLvl + \$FighterLvl + \$WizardLvl))$$

The player can now choose between those 2 options:

- Attack
- Heal (cannot go above 8HPs max)

Attack damages are calculated as follows, depending on the player fighter skill level:

$$\$Damages = 1 + rnd^3(\$FighterLvl - 1)$$

HPs healed are calculated as follows, depending on the player healer skill level:

$$\$Heal = \$HealerLvl + rnd(max^4(0, \$EnemyHP / 2 - \$HealerLvl))$$

If the enemy is still alive, one random option between those 2 is chosen (calculations are right below, respectively):

- Attack
- Heal (cannot go above 8HPs max)

$$\$Damages = 1 + rnd(\$FighterLvl - 1)$$

$$\$Heal = rnd(\$HealerLvl)$$

If the player is not dead, the combat turn restarts.

---

<sup>3</sup> Random integer from 0 to the parameter in parenthesis

<sup>4</sup> Maximum number between the two parameters in parenthesis



## Experience And Leveling

For each damage dealt / HPs healed, 1 XP<sup>5</sup> will be awarded.

Base class level is 1 and have no cap. However, it will be displayed until level 7 as the game ends most of the time before.

After level 7, the display gets back to one as the handling switch's default case is set to display the first frame.

The player needs to inflict 4\* <class level> damage to level up his fighting skill.

The player needs to heal 3\* <class level> HPs to level up his healer skill.

Extra XPs are stored.

Note: enemies are scaling based on the player's level, taking every fight is not a good idea...

---


<sup>5</sup> Experience (points)

## Levels / Screens


### Assets

### Scenes










Those screens have been generated by IA and reworked with PixelLab extension for Aseprite.

<i>Splash Screen</i>	
<i>GameOver Screen</i>	
<i>Victory Screen</i>	

## Tilemap

Tilemap	Tilemap (annotated)
	

## Sprites

Player (contain idle / walk animations; 2 frames, for each direction)	
Enemy Variation 1: 2 frames	
Enemy Variation 2: 2 frames	
Chest: 2 frames	
Door: 2 frames	
Locked Door: 2 frames	
Health Bar: 9 frames composed of those sprites	
Skills level: 8x2 frames composed of those sprites	
Boss: 2 frames Note: Not displayed correctly in the game for a reason I don't know	

## Sounds / Music

Splash Screen	"Lands with no Dragons DX".mod
In Game	"Unkown Caverns DX DRUMLESS SLOW".mod
GameOver Screen	"Ruin at last DX".mod
Victory Screen	"Crystal Clear DX SLOW".mod

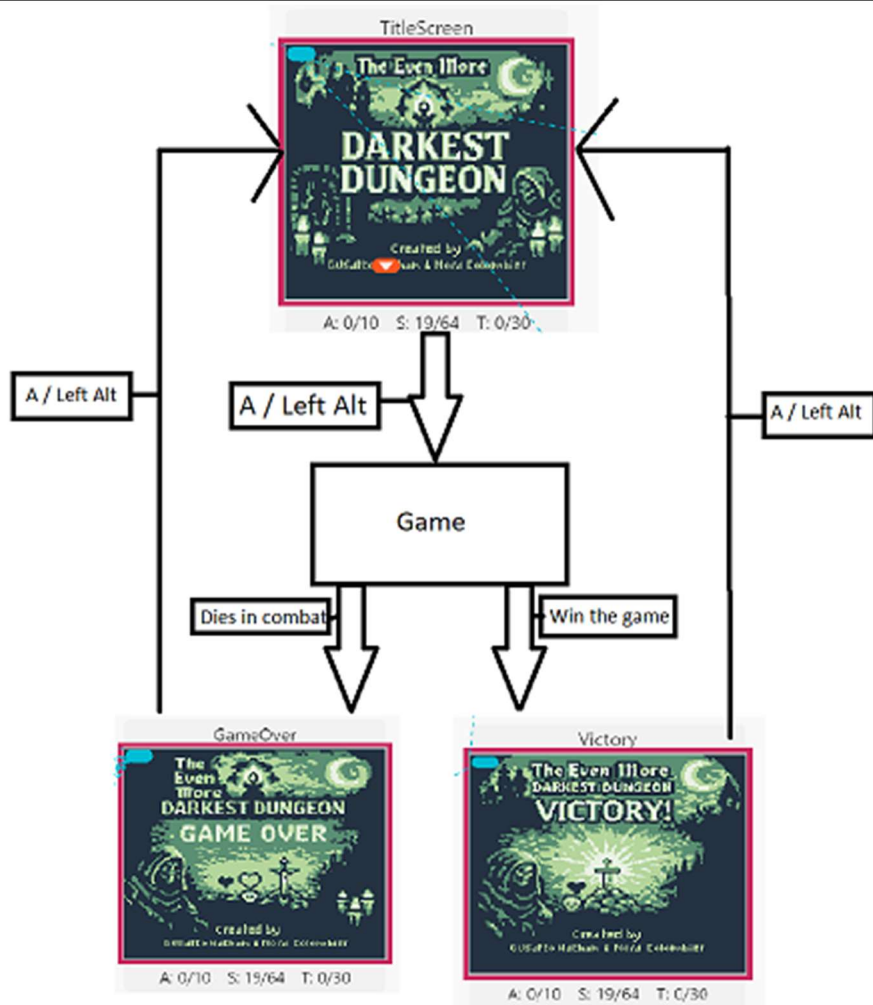
For SFX, I use a SFX array (vyskeisfx1.sav) made and described (vyskeisfx1.txt) by the creator.

Here is the index of the SFX I use in each case:

Enemy / Player death	25
Enemy hit	20
Enemy / Player heals	18
Player hit	21
Door / Chest opens	13

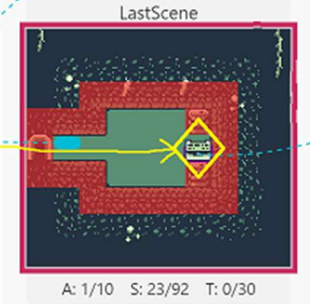
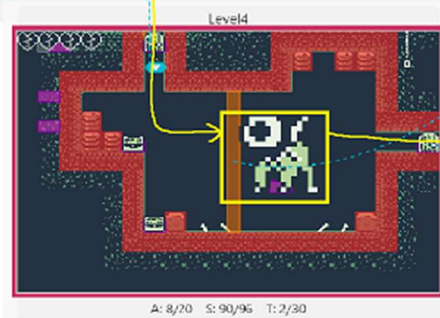
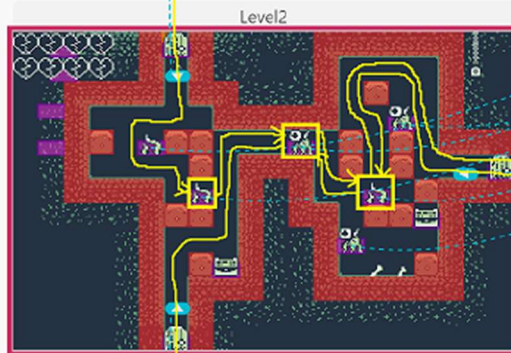
## Game Flow

### Main Flow



## Level Flow

- Critical Path is displayed in yellow
- Yellow Circle is the player start
- Yellow Squares are obligatory fights
- Yellow Diamonds are obligatory chests



Level 1



A: 12/20 S: 54/65 T: 1/30

Level 2



A: 15/20 S: 54/96 T: 3/30



Level 3



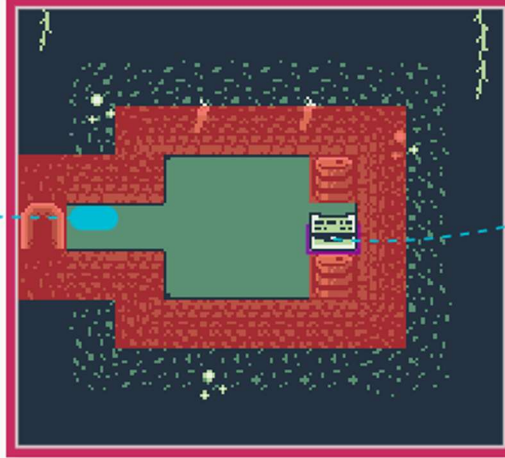
Level 4





Last Scene

LastScene



A: 1/10 S: 23/92 T: 0/30

## Critical Path / Design

To win the game, the player needs to travel through levels (as yellow path shows: cf. [Level Flow](#)). The player at least needs to fight enemies on his path which are symbolized with yellow squares and open at least chests that are symbolized with yellow diamonds.

Optional enemies and chests are available to level up players' skills / or heal the player.

Other chests grant the player +1 HP when opened.

The first chest on the player path contains the key to access the "Last scene" and therefore the victory. The player can't access it without the key.

The second chest on the player path (and the last chest) is the chest in which the player gets the sword and finishes the game. After the short dialogue, the player gets teleported to the victory screen.

The level design requires the player to go at least once to every level, and depending on the outcome of fights, the player may need additional fights to collect extra HPs and XP to be strong enough to beat the boss.

To complete the game, the player needs to have at least 7 HPs to beat the boss.

Because of chests presence around the boss and the scene before, the player can finish the last enemy before the boss with at least 4 HPs.

Because of a bug<sup>6</sup> which reinitializes the scene when the player goes to a different scene and gets back to this one, the player simply needs to be alive at the end of the last fight and then use the bug to regenerate his life up to 7 HPs before the boss.

This is a known bug but because of constraints in memory it is not solvable yet without cutting content in the game.

Only the boss state is stored in the memory. Other enemies / chests respawn each time.

---

<sup>6</sup> There are a few technical bugs but unfortunately, they are not all solvable because of technical restraints in memory, graphism...

## Bugs / known issues

Here are the bugs and issues that are known and can't be solved because of time / technical constraints:

- The boss is not displayed correctly (the sprite seems cropped)
- There is one frame missing in the player's animations
- As listed before, the player can exploit the game by resetting the actual level and restore his life entirely
- Attack damages of enemies are not capped so they can simply one-shot ( $\geq 8$  damages) the player if he scales too much (because the enemy's attack damage scales with the player's fighting skill)
- Skills levels are well displayed until level 7, and then the display gets back to 1 because of the use of a switch and the default case is level 1.
- Because of graphical constraints and number of actors displayed at the same time, certain actors tend to disappear during frames in combat.
- Graphical layers tend to fuse themselves and make graphics display on top of others.

# Appendix

## Glossary

- HPs: Health Points
- D-Pad: Directional Pad
- Rnd: Random
- Max: Maximum
- XP: Experience

## Tools / Softwares

Those tools were used during the make of the game:

- GB Studio (Game Engine)
- Aseprite
- PixelLab extension for Aseprite
- Generative IA

## Credits

- Nathan Gusatto: Programming and making of the game.
- Nora Colombier: Artist
- TipTopTomCat: Music (<https://tiptoptomcat.itch.io/8-bit-gameboy-songs-gb-studio>)  
*"All songs are original compositions by me (TipTopTomCat) unless otherwise noted. All songs in this series were made using OpenMPT software. These assets can be modified and used for commercial or non-commercial projects but can not be sold, uploaded to streaming platforms, or claimed as your own work."*
- Vyskei: SFX (<https://vyskei.itch.io/vyskei-sfx-1-gb-studio>)  
*"You can use these sounds for whichever project you like whether commercial or non-commercial, so long as you are not reselling the raw assets themselves. I don't require credit for use, but you may if you like."*
- Generative IA:
  - o Splash Screen
  - o Game Over Screen
  - o Victory Screen
- Special thanks to the random guy who commented the game on itch.io, it made my mood go brr.