



# Dawns Gate

## Project details

Game design and development MSCs

Mathieu Schmerber

15 AUGUST 2022

## TABLE OF CONTENTS

<b><i>Game design concepts</i></b> .....	<b>2</b>
Overview .....	2
Character, Controls, Camera .....	2
Combat .....	3
Run concept .....	5
Bestiary .....	8
Scaling .....	10
<b><i>Project management</i></b> .....	<b>12</b>
Team .....	12
Sprints .....	12
Play testing .....	13
<b><i>Artistic direction</i></b> .....	<b>14</b>
Influences .....	14
Visuals .....	14
Musics and sounds .....	15
<b><i>Game development</i></b> .....	<b>16</b>
Optimisation .....	16
Development Tools .....	17
Additional features .....	17

## GAME DESIGN CONCEPTS

### OVERVIEW

*Dawns Gate* is a 3D isometric combat-focused Roguelite. It takes inspiration from popular games in the genre such as Hades, Skul the hero slayer and Dead cells.

*Dawns Gate* seeks a nervous gameplay, which heavily relies on the player reflexes and the learning curve. It is meant to be a hard game to master, as expected of a Roguelite. This game suits players who like to grind and trial-and-error based games.

Although the gameplay doesn't tell any particular story, *Dawns Gate* still defined a short lore for the sake of game design and artistic direction consistency.

---

### SHORT LORE

It is at dawn's gates that since the beginning of time and for eternity, an umpteenth battle of heavy consequences and anticipation is being prepared. This very battle between day and night, which results alone, might present another glorious sunrise to humanity.

Within an unfathomable world far from humanity's grasp, Solaris' and Lunaris' armies have been trying for immemorial times to enslave each other. These repeated battles have established a fragile balance that we human beings, naively call "day-night cycle". Only when the Solaris' army loses ground does the Lunaris' army advance, leading to nightfall.

We, human beings, wouldn't possibly know that a single defeat from Solaris against his antonym could plunge humanity into the most eternal of nights.

---

### CHARACTER, CONTROLS, CAMERA

---

#### CHARACTER

The player plays Solaris. As explained within the lore, Solaris is an ominous monk representing the sun.

As a persona and by design, Solaris should not look like a friendly figure. Solaris is ominous, the player should be thinking of Solaris as a non-human entity with an unconcise goal.

---

## CONTROLS

The game supports mouse and keyboard controls as well as any type of controller. The choice of one control over another does not provide the player with any advantage.

I would still recommend the use of a controller over the mouse and keyboard as I feel like it is more suited and feels better to play, however this is only my opinion.

---

## CAMERA

The camera is described as a top-down 45 angle isometric camera. This design choice provides better gameplay readability. Since Dawns Gate seeks a more skillful gameplay based on combat and movement, the camera should help the player better understand what is happening on the screen.

The camera should never be penalising the player, that is why we are removing the perspective with the isometric setting. Getting rid of the perspective, better defines areas of effect and the position of each entity or threat on the screen.

## COMBAT

The core of the combat system relies on two main mechanics: Attacking and Dashing.

---

## ATTACKING

Dawns Gate combat is simple yet engaging. It is designed to be easily understandable by any player and could be compared to a simple “mashing button” mechanic.

The combat system consists of creating attack combos, a combo is a mix of two light attacks and a final heavy attack closing off the combo. To execute the combo, the player only needs to consecutively press the attack button.

What makes the combat system challenging is the heavy attack. A heavy attack will apply more damage than a light attack while being much slower to execute than the other type. This makes the attack rewarding to hit, as the hitbox of this one is also bigger than that one of a light. The slow pace of a heavy attack makes it also situational since missing this attack could lead to the player being punished by surrounding threats or enemies. The player must constantly weight the opportunities and threats of executing a heavy attack every moment.

In combos, not executing a heavy attack also means stopping the combat pace for a moment to break the ongoing combo. This is how Dawns Gate achieves a simple yet engaging combat mechanic while being innovative when looking at other games of the genre.

---

## DASHING

While the combat system is innovative, the dashing is pretty common within the Roguelite genre. Dashing provides the player with extended movement options. The dash can be used to either dodge an attack or extend an attack within a short range. In Dawns Gate, the dash can be used every 1.5 second by default and provides with a few frames of invulnerability. This duration may become shorter during the game according to the scaling system.

The dash cooldown is short since we want to emphasize the nervousness and fast paced combat of Dawns Gate. After playtesting the game, it appears that dashing at the right timing is crucial and consuming it too eagerly might result in the player being heavily punished, even with the dash cooldown being so short.

---

## WEAPONS

The player can choose between three weapons to equip, before starting a new run:

### **The Sword**

A very quick but weak weapon with 2 quick light attacks and a thrust, as a heavy attack. The light attacks have a decent range but small area of effect, while the thrust is slim but incorporates a free dash. This weapon is excellent at targeting a lonely enemy, while being safe.

### **The Staff**

A slow weapon with heavy blows. The two light attacks have short range but a 180-degree area of effect, while the heavy attack is a ground slam dealing heavy damage in a large area but sticking the player in place for a breath. This is a great weapon at dealing with many enemies, it is dangerous to attack but very rewarding.

### **The Scythe**

The middle point between the two previous weapons. A good range and area of effect on the two light attacks, but a very short range on its heavy blow. The heavy attack is a 360-slash around the player, which is really easy to hit but also means to take greater risk to hit more enemies in one blow.

## RUN CONCEPT

---

### GENRE DEFINITIONS

With the Roguelite genre comes a few must-know definitions and concepts.

#### Run

A run is a set of **Rooms**. Runs are usually randomly generated with pre-constructed rooms, but might also generate each room procedurally or randomly. The run generation rules are defined by the game and usually differ from one game to the other. A run ends whenever the player dies or when defeating the last room (usually containing the final boss of the game). During a run, the player might collect items, spells or whatever object to help him beat the game. Dying in the run results in the player losing all its progression, making each run unique.

#### Room

A room is a part of a run and can consist of anything. A room could contain simple fighting events or define a unique mechanic depending on the game. A room could be a small platform or a whole world on its own.

#### Lobby

The place where the run starts. Usually, this place is used for the player to rest in-between runs, and gear up. In Dawns Gate, the lobby is where the player gets equipped with one of the three available weapons.

---

## RUN STRUCTURE

In Dawns Gate each run is a set of 15 rooms. The run is randomly generated from pre-built rooms. The run is generated according to a custom set of rules, which affects the likelihood of some room types over some others.

The room types are listed as follows:

- Combat rooms (9/run on average)
- Shop rooms (2/run average)
- Life shop room (1/run average)
- Event rooms (2/run average)
- Boss room (1 per run)

Each room is a small platform, usually the player can fit the whole room within the camera bounds. This allows the player to be aware of everything happening while fighting, which again helps Dawns Gate achieve its goal of skilful and fair gameplay.

Combat rooms are basic rooms where 2 to 3 waves of monsters spawn upon entering one. A wave of monsters consists of 4 to 6 monsters.

---

## SHOPS AND LIFE SHOPS

Shops are a way for the player to scale during the run by buying items. During the run, the player might collect gold to spend in shops.

Gold can be used to buy items or upgrade them (see the run scaling section). The player can also find a statue of the sun to praise. Praising the statue provides the player with health regeneration.

The player might also come across a life shop. Life shop allows the player to buy items in exchange of a portion of its maximum health. Items purchased in life shops are usually stronger than the average item due to their cost. It is impossible to regenerate any health or upgrade any items from within the life shop.

---

## EVENT ROOMS

Event rooms are pretty rare within a run and are meant to provide variety to this one. There are 3 different event rooms in Dawns Gate. A full run will only feature 2 maximum event rooms, this creates diversity in-between runs as well. A player having cleared a full run, won't necessarily know about every feature of Dawns Gate.

### The Casino

The Casino is a passive room with no fighting . The player will meet The Croupier, an NPC, which will ask the player to play a game. The game is a bet, which can be refused by the player without any issue.

The game consists of betting on any item or gold randomly asked by The Croupier. When placing the bet, the player has a 50% chance of winning it. Upon winning a gold bet, the player wins 200% of its bet. Winning an item bet will upgrade the player's bet. Losing the bet will result in the player losing what was placed.

### The Time room

The time room is similar to a combat room. The room consists of fighting on top of a giant clock. The player will need to trigger a magic hourglass at the middle of the room. Upon activating the hourglass, the clock will begin to tick. While ticking, monsters will spawn on the clock without rest. When the minute hand on the clock completes a full turn, every monster die. The time room is essentially a "survive a whole minute" room.

### The Gobbling Dealer room

This room features 4 pillars and 4 walls around it. In the middle of the room, sits an NPC similar to a cook. The NPC will ask the player to make a deal at random, either an item for golds or golds for an item. The deal will not necessarily be affordable.

Accepting an affordable deal will clear the room. Refusing or not completing the deal will enrage the NPC, turning it into the **Gobbling Dealer**. The Gobbling dealer is an aggressive mini-boss, and the mechanics are explained later within the "Bestiary" section.

---

## BOSS ROOM

The Boss room is the last room of each run, and can be attained by the player as long as it doesn't die prior to entering it. Upon entering the room, the final fight between Solaris and Lunar is starts. Lunar is a boss, its mechanics are explained within the next "Bestiary" section.



## BESTIARY

---

### SEEKER

The seeker is a simple monster that spawns in combat rooms. It has a passive AI, which will only roam around the room, until the player enters its range. Upon detecting the player nearby, the seeker will dash through the player and any other object in the room after charging up.

Touching the seeker in dash animation will apply damage and a slight knockback to the player.

The player must wisely use the dash mechanic to dodge the seeker's dash.

---

### TANK

The tank is a simple monster that spawns in combat rooms. It has an aggressive AI which will chase the player upon detecting its presence nearby. The tank will slam the ground, applying damage in a small area of effect. The only way of applying damage to the tank is to hit it when it is attacking, otherwise the tank will block the incoming attack. When successfully blocking an attack, the tank will be provided with a portion of armor, which sits on an additional health bar, effectively prolongating the fight.

The player needs to wisely use the dash mechanic to dodge the tank and punish it while still in the attack animation.

---

### THROWER

The thrower is a simple monster that spawns in combat rooms. It has a passive AI, which will only roam around the room, until the player enters its range. Upon detecting the player nearby, the thrower will launch a fireball at the player. The fireball will leave a preview of the area of explosion for the player to dodge. The fireball explosion will leave a fiery area on the ground for a few seconds. Walking within this area will apply a burn effect on the player, damaging it over time.

The fireball explosion area can be dodged without using the dash mechanic. The remaining area is used to restrain the player's movement freedom.

---

### GOBBLING DEALER

The Gobbling Dealer is a mini-boss with one phase and two attack patterns. During the whole fight, the Gobbling Dealer is assisted by 4 lighten up totems, making him invulnerable as long as at least one of the totem is still bright.

---

For his first attack, the Gobbling Dealer must teleport to the center of the room. After the teleportation, he will start casting 4 lasers, each executing a 360 degree rotation for a few seconds. The only objects capable of blocking the laser are the walls present within the room. The Gobbling Dealer stays completely still during this attack, which makes it an easy target if the invulnerability worn out at some point. The player must take cover behind the walls to avoid any damage, and can also dash through the laser if necessary.

The second attack consists of the Dealer dashing 4 to 6 consecutive times across the room, from one extremity to the other, whilst aiming at the player. Dashing through the player applies a fair amount of damage to the player. Dashing through a totem will extinguish it for around a minute.

The Gobbling Dealer constantly rotates through these two attacks until it dies or kills the player. The player should learn the correct timing to dodge the Gobbling Dealer dashes while directing them at the totems. The player needs to be very efficient at dealing with the totem, as these totems might light up later, especially while the Dealer casts its lasers and is unable to deal with the totems for the time being.

---

## LUNARIS

Lunaris is the final boss of Dawns Gate, he has 3 different phases with 3 attacks per phase. Lunaris design is close to Solaris', making it the only entity looking like the player character. Just like Solaris, Lunaris can equip a sword, staff and scythe, the difference being that Lunaris will switch between his weapons during the fight.

During the whole fight, lightnings are constantly spawning at random locations nearby the player, to restrict its movements and disturb its focus. This room passive will scale on each new phase, making the dash mechanic especially relevant.

For the first phase, Lunaris equips the scythe. Lunaris has the same attacks as Solaris does, but with at least two times the range. During the scythe phase, Lunaris can dash to extend his attack range. The room lightnings spawn every 2 seconds near the player.

The second phase is the sword phase. While on the sword phase, Lunaris sword thrust makes him dash to the nearest wall, bumping its body into it, which results in Lunaris being stunned for a few breathes. While in the sword phase, more lightning are spawning, to prevent the player from attacking too much during the stun frames.

The third phase is the staff phase. During this phase, the lightnings become a real threat to the player. Lunaris's slam will create a shockwave growing bigger and bigger overtime until it covers the whole room. The player will need to time its dash to dash over it without being hit.

## SCALING

---

### RUN SCALING

---

#### ITEMS & INVENTORY

During the run, the player will come across items that will make Solaris stronger. These items can be looted randomly upon clearing a room or bought in shops. The items can be categorised in three types:

**Stat items:** Provides fixed stats (example: +10% attack damage)

**Active item:** Triggers an effect after executing an action (ex: spawns lightings when killing an enemy)

**Passive item:** Triggers an effect independently (ex: regenerate 2% of health every 10 seconds)

The player will put items in its inventory in order for these to affect it, however the inventory is limited to seven different slots; the player will therefore need to make wise choice as to what items to keep and throw away. The player cannot have the same item equipped twice and will never encounter an already equipped item.

---

#### UPGRADING AND MERGING

To make up for the inventory limitations, items may be upgraded or merged in shop rooms for gold coins. An item can be upgraded twice, making its effect stronger.

Merging two items together will combine the two effects into one item. This action is wise since it essentially frees one of the inventory slots. However, there are some limitations to the merging mechanic. An item combination can only occur if at least one of the two items is a stat item, and a merged item cannot be upgraded anymore. Merging an item is therefore a definitive action which shouldn't be performed on the fly.

---

#### SCALING STRATEGY AND THEORYCRAFTING

Dawns Gate features 35 different items, taking the inventory limitations into account, this allows for different items builds to be constructed and theory crafted, especially as some items won't be suited for any given weapons. If the player is equipped with a sword, items providing more damage on wide attacks will not be as relevant as if the scythe is equipped for example.

As mentioned the items can be looted at random upon clearing a room, which would make the theory crafting very random. However, in shops, the player is free to choose between 3

different items to buy. This means that the player should collect enough gold to be able to choose its items.

Clearing rooms may spawn an item or provide the player with gold. After clearing a room, the player can choose between multiple doors to enter the next room. Each door displays its room outcome visually, for example, doors with a giant gold coin at the top will lead to a gold room. This means that the player is always free to choose the room outcomes during the run. The player can either target instant growth by entering an item room, despite the randomness, or might choose to harvest as much gold as possible and get to choose and upgrade items.

This "choose and collect" mechanic, combined with the inventory limitations and the merging system, creates great theory crafting and equipment strategy during the run itself !

---

## LOBBY SCALING

The concept of run itself makes it impossible to save progress between two runs. However, in Dawns Gate, it is possible to make Solaris scale permanently. While clearing room, Solaris harvest what's called Souls, which is an additional currency of the game. Clearing a room provides the player with a small portion of soul, while killing Lunaris provides it with a greater amount. These souls can be spent within the Lobby on character traits.

Character traits are permanent stats upgrades, such as increasing Solaris maximum health or damage. Traits can also increase Solaris gold income after clearing rooms, or even get Solaris the revive ability during a run.

## PROJECT MANAGEMENT

### TEAM

I, Mathieu Schmerber is the one and only author of the game design and development of Dawns Gate. With the agreement of Stefano Padilla, two of my friends are helping me complete the project, regarding artistic parts of the project only. The team can be presented as follows:

#### **Mathieu SCHMERBER**

Game design, game development, visual effects, 2D Art and UI, Sound design

#### **Philémon Carron**

3D Art, Visual effects

#### **Quentin LE VAN**

Music theme composition

### SPRINTS

I organised the game development into a set of sprints and user stories, as I would do for any Agile SCRUM development. Everything has been laid out on Hack'n Plan, a project management tool online made specifically for game development. Sprints can be summarised as follows:

Sprint name	Due date	Allocated time
Combat system	June 6 <sup>th</sup> 2022	1 week
Run management	June 13 <sup>th</sup> 2022	1 week
AI system & behaviour	June 20 <sup>th</sup> 2022	1 week
<b><u>Combat game feel playtesting session</u></b>		
Run scaling	July 4 <sup>th</sup> 2022	2 weeks
<b><u>Combat and scaling playtesting session</u></b>		
Lunaris	July 11 <sup>th</sup> 2022	1 week
Event rooms	July 18 <sup>th</sup> 2022	1 week
<b>End of game play milestone</b>		
Game polishing & playtest	August 7 <sup>th</sup> 2022	2 weeks
Final art integration	August 15 <sup>th</sup> 2022	1 week

Most of the sprints were purposefully overestimated for risk mitigation purpose. The risk mitigation took into account potential motivation burnout and massive bugs encountering. Thankfully, no major bugs occurred throughout the development, which led me to have over a week worth of ahead time near the end of the project, which was partially allocated for additional game polishing and to take a few days of break as I began to feel tired and didn't want to burn out my motivation.

## PLAY TESTING

I organised two different playtesting sessions publicly with my friends, external to the project. Dawns Gate needed some major feedback on the combat mechanics and game feel rushing in since this is the selling point of the project.

The first playtesting session lacked game-feel on the attack system, which didn't feel "juicy" enough at that time. The feedback from the first session allowed for the combat system to feel better, adding screen shakes and controller vibrations to make the player feel every heavy impact. The playtesting also showed some really unfair balance between the weapons, which was quickly adjusted thanks to the development tools (see the "Game development" section).

The second playtesting session occurred in a state of the game where the game feel was much better, and when the items and run scaling mechanics were just introduced. At that time, there were 20 stat items within the game, and 3 of them were marked as not interesting enough. These items were later replaced by 3 additional active items, which generally speaking brings more fun to the combat system as more effects appear on the screen while using them. During this session, some players felt frustrated with the dash mechanics. Initially, performing a heavy attack prevented the player from dashing to emphasise the danger of thoughtlessly rushing in. This mechanic felt unfair to players using their heavy blow carefully since they were still getting punished because of the animation time. To resolve this issue, the dash became an ability which can be triggered whenever the player wants, even when stuck in some attack animation.

This dash change made the game feel way more responsive and rewarding, the player now feels in control over the situation.

## ARTISTIC DIRECTION

### INFLUENCES

Given the time limitations of the project and availability of Philémon Carron, we have chosen to go for a stylised artistic direction. This style doesn't require high polygon models and is usually a perfect match for isometric games since the art must comply with the game design aspects. Philémon initially wanted to create a dark fantasy world under the influence of the infamous Dark Souls games and Elden Ring, which in contrast, requires very high-quality models.

We decided to try merging the dark fantasy and stylised directions, which led us to the style of Tim Burton.

### VISUALS

---

#### 3D ART

With the influence of Tim Burton's *The Nightmare Before Christmas*, we tried creating very organic art, where every object could be twisted. However, the isometric game camera limitation was hard to counter balance as the twisting effect required some kind of perspective.

The design of the important entities of the game was made to be somewhat goofy given the stylised art direction, but we still needed to find a balance between the goofiness and the dark fantasy we were looking for. We ended up creating very ominous figures for Solaris and Lunaris, putting masks on chibi-like characters. Since NPCs were not wearing masks, we designed them to be as creepy as possible.

---

#### 2D ART AND USER INTERFACES

I decided to draw the game's items in 2D to create an interesting contrast with the 3D models. The 2D art needed to fit with Tim Burton's aesthetic as well, which results in very rough brush strokes and clear line-work imperfections. Once again, the game camera being zoomed out turned out to be challenging to design the items as the organic line art were too detailed for the game view. I went for an art style where the line art is incredibly thick and rough, and decided to create an organic feeling by emphasising the colouring and shading of each item. I essentially created all items with a very thick outline, and absurd light and shadow effects. The in-game results are convincing, as long as the items are not zoomed on.

For the user interfaces, I could freely use the first line art method, which was not working for the items, as the art pieces could just be displayed on the screen directly. I decided to create as few UI as possible as Dawns Gate is focused on combat, therefore should not include a heavy menu-ing.

## MUSICS AND SOUNDS

---

### THEME CHOICES

The music themes of Dawns Gate are heavily influenced by the dark fantasy genre. We wanted to maintain a dark but calm mood throughout the run to emphasise the sound design of the combat. In this way, it is the player who creates the rhythm of the combat, on calm and neutral background music.

The background music theme allows us to surprise the player sometimes with a sudden theme switch. The clearer example of a theme switch is when entering an event room. Event rooms are meant to create diversity within the run, so switching music just for event rooms is a great way to create the surprise effect. For example, when entering the Casino and meeting the Croupier, the music switches out from very dark and cold music to a happy and jazzy theme, which breaks the whole mood of the run during just a bet. We can also see this effect in the Deal room. Upon refusing the trade, we are switching to a mix of dark fantasy and an electro house styled track to surprise the player with an unexpected boss.

---

### SOUND DESIGN

Dawns Gate heavily relies on its sound design. As previously mentioned, the lack of UI combined with the very neutral main theme, allows for great immersion within the gameplay. This immersion can be created thanks to subtle sound effects such as subtle footsteps with different tracks depending on the material the player is walking on, or more obvious sounds such as sword slashes, fire crepitating, ...

The sound design and music create a very subtle and delicate volume balance, which needs to be adjusted in real-time. To resolve this issue, an audio menu was created in the game, for the player to tweak itself.



---

#### ARCHITECTURE INSIGHTS

Dawns Gate was developed with scalability and optimisation in mind. Owing to its roguelite nature, the architecture of the game involves many systems and subsystems that are well structured and optimised. For example, while in runs, the combat system will constantly be called. The combat system will need to call over the weapon system for each combo, each combo will need some relationship with the player statistics system and each combo attack will need to keep the inventory system in check in case any item of the item system must register an action. These multi-system architectures can quickly build up dozens of relationships for each major call. While having many relationships might create some concept complexity, as a developer we must primarily ensure that relationships do not become dependencies as we want a system to be as independent and generic as possible.

Since the beginning of Dawns Gate, a lot of time was allocated to design a stable architecture to avoid dependencies and unnecessary memory allocations. This is also the main reason why no major bugs were encountered during the development. To avoid dependencies, I chose an event subscription-oriented development pattern, which despite being slower to process than basic dependency relationship calls, results in every system and subsystem being fully independent.

---

#### OBJECT POOLING

Since Dawns Gate must spawn in many objects during runs, I chose to use object pooling to minimise the resource cost. The concept of object pooling consists of allocating memory for instances, and returning it to some pool when the instance is of no further use. This way, for each instancing, the game can just look for already allocated memory within the pool.

To provide some numbers, we can estimate the number of enemy spawns to be around 180 per run. With a basic instantiation, the game would dynamically instantiate 180 times per run, only taking enemies into account (12 per room for 15 rooms). If we consider that each enemy only takes 3 hits to be defeated, we can also add an average of 8 VFX instantiations per enemy (Attack visual, hit visual, death visual, potential item active effect visuals, ...). This average would lead to around 900 instantiations per run, which doesn't even consider half of the required instantiation of the game, as we also must spawn in enemy attack, items, text effects,...

Object pooling quickly becomes a must-implement optimisation feature if we were to aim for more than a prototype. In fact, given the previous numbers, the minimum instantiation number required for the whole combat system to work would drop down from 900 per run to 48 per game launch since a wave is composed of 6 monsters max.

## DEVELOPMENT TOOLS

Given the time limit of the project, Dawns Gate needed to create the most content out of the fewer game design concepts possible. This amount of content can really affect any side of the game, from items and weapons to enemies and even bosses. It quickly becomes obvious that it is not feasible to code every object one after another.

That's why I chose from the earliest stage of development to start developing tools which would only be used for development. This way we now have an arsenal of content creation tools available to us. We can essentially create any type of new content from within the game engine without the need of adding a single line of code to the project. That's how I could create 35 different items for the game, of course, I did not code every item one after the other.

Although developing tools is time consuming, in the end, it proved to be really worth it, especially after playtesting sessions, as Dawns Gate required a lot of balancing. Having these tools made the balancing really straightforward, essentially tweaking each value without causing any issue within the code base !

## ADDITIONAL FEATURES

The main features of Dawns Gate were mentioned throughout this entire game design document, here are the additional features that could not be described in detail:

- Main menu
- Settings menu
- **Save system** (saving souls amount, character traits, audio balance, ...)
- **Combat effect system** (burn effect, curse effect, ...)
- **NPC Dialogue system** (talking with The Croupier or The Dealer with complex deal logic)
- Scene transition system (multi scene workflow optimization)