

#### IN THIS CHAPTER

- » Learning what game design is
- » Thinking like a game designer
- » Overviewing how game design became more accessible
- » Leaning into the fun
- » Making your first game

## Chapter **1**

# Introducing the Art of Game Design

I'm so excited for you to dive into this book, break down the mysteries of game design, and maybe even start working on your own game. Maybe you're just getting started or maybe you're already deep into a project. Either way, this chapter can help you understand what game design really is, how it's evolved, and how you can use it to design your games.

## Discovering What Game Design Is

When I think back to all the work I've done on games and the work I see my colleagues doing, it's clear that being a game designer means wearing a lot of hats. Sure, most jobs require a mix of skills, but a game designer is kind of like the general manager in construction. You need to know a little bit about a lot of different things. And the things you need to know differ from genre to genre. Because every genre has its unique elements, many game designers specialize in one area, like level design or progression, and build their expertise from there.

Game design is making a thousand little decisions that add up to one amazing experience, like the following:

- » Tweaking how many items drop from a chest
- » Deciding why some items are rare and making them feel special and hard to get
- » Designing levels that guide players without them realizing they're being guided
- » Balancing abilities so that no one character or strategy dominates the game
- » Writing lore that makes players care about the world and its characters
- » Mapping out progression systems that keep players coming back for “just one more level”
- » Deciding how fast a character runs and how high they jump
- » Choosing the perfect sound effect for picking up a coin
- » Testing and re-testing mechanics until they feel “just right”
- » Creating tutorials that teach without boring or frustrating players
- » Ensuring that the game is accessible so everyone can play and enjoy it
- » Crafting choices that make players stop and think about the consequences



REMEMBER

When someone takes on the role of a game designer, they are responsible for orchestrating all these decisions, building frameworks to guide them, and crafting systems that streamline the decision-making process. See Chapter 4 for more about what it takes to become a game designer.

## Looking at a Brief History of Game Design

I frame this section around notable moments in history when game design and development became more accessible and how these changes shaped the way we make and play games, working back from today. I stick to the last 30ish years or so, just because the internet became mainstream and available in many U.S. households in that time frame. Ever since then, games have been growing at an exponential rate.

As of 2025, it's incredibly simple to get started in game design. You can download game engines like Unity, Unreal Engine, or Godot for free access resources like free asset libraries, and learn new skills through platforms like YouTube.

Software like Blender (a free tool for 3-D modeling) or Procreate (for digital art) allows creators to produce their own assets, and platforms like Itch.io let anyone publish games for free, often within minutes. And once you publish a game, social media lets you promote it to people who may want to play it. The barriers to entry for game design today are time, practice, and dedication rather than expensive tools or access to industry secrets and publishers.

The result is an explosion of creativity and people who publish games:

- » More than 89,000 games were published on Steam as of January 2025.
- » Thousands of genres and user-generated tags describe a diverse array of experiences, from “frog detective mysteries” to “farming horror simulators.”



REMEMBER

The democratization of tools and knowledge enables everyone to shape the industry, instead of a select few dictating what games are. Just like you can take many different paths to make a game, you also can define what a game is in many different ways. At the time that I’m writing this, the industry is having a difficult time. Many companies are doing layoffs, and larger projects are having trouble getting funded. But making games on your own and sending them off into the world has never been more accessible.

Time to go back in time, starting with 2016:

- » **2016:** *Pokémon GO* was released, which was a global hit. This game took advantage of what mobile phones have to offer like most other games hadn’t: portability to explore the world and their camera. Ingress, the game made by Niantic prior to *Pokémon GO*, has done this as well, but adding in Pokémon was lightning in a bottle for Niantic. For me, *Pokémon GO* inspired me to focus my graduate work on augmented reality (AR) and map-based mobile gaming. I eventually got to work with some of the people who made the first version of *Pokémon GO* when I designed *Peridot* at Niantic, a fortunate time in my career.
- » **2014:** Facebook (now Meta) acquired Oculus, making VR a mainstream focus for tech and gaming. This acquisition has since really changed the VR landscape! For example, I was in a Target the other day and walked past an aisle end cap holding new Quests for sale. You never would have seen that in 2014. Alongside their hardware efforts, tools like Horizon Worlds and other accessible VR platforms are making it easier for smaller teams to experiment with VR and AR game design with a built-in audience.
- » **2013:** The launch of the PS4 and Xbox One gave indie developers easier access to console publishing thanks to digital storefronts like the PlayStation Store and Xbox Live Arcade. However, to this day, some friction exists in publishing

on consoles due to factors such as platform-specific certification processes, higher development costs for console-ready builds, and strict licensing agreements. Additionally, console publishers often require developers to secure development kits, which can be expensive or difficult for people to access.

- » **2010s:** Kickstarter, a crowdsourcing platform for new products, took off. This enabled board game designers and video game designers to post their game online and get some funding to work on their game. Big games like *Cards Against Humanity* came from Kickstarter. And it wasn't until 2012 that Oculus launched its first Kickstarter for the Oculus Rift, sparking widespread interest in virtual reality (VR) gaming.
- » **2009:** People became more aware that a small team or even a solo developer could release something that could become huge when they saw *Minecraft*. The developer, Notch, became a great example of a solo developer who achieved global success with a release on the internet.
- » **2007:** The launch of the iPhone and the App Store introduced a new era of mobile gaming. It wasn't until 2009 that some mobile games become phenomena like *Angry Birds* and *Fruit Ninja*. Mobile phones also introduced the free-to-play (F2P) model, allowing developers to monetize games through in-app purchases and ads, which is now big business.

Also in 2007, Facebook opened its platform to developers, leading to the rise of social games like *FarmVille* (2009) and *Mafia Wars* by Zynga. These games pioneered new ways of engaging players through social mechanics and microtransactions.

- » **2003:** Valve did something really big and launched Steam. This made it easier for indies to release their games alongside all the larger studios out there. Back in those days, there were way fewer games being published on Steam. A year after release in 2004, only 65 games were published. But, obviously, that took off between then and now!
- » **1990s:** Last, but certainly not least, the internet became accessible to lots of households in the United States. Dial-up became the standard for home internet connections, using telephone lines to connect to the internet. This enabled games to be created and shared with people outside of your household without a publisher. Dial-up began to decline in the 2000s with broadband internet, but I'll never forget all the times my friends were over at my house playing some game we found on the internet, hogging the phone line from my parents.

Trekking back to today, top genres of games include open-world role playing games like *The Witcher* and cozy games like *Stardew Valley*. There's been a rise in

more niche genres in the last few years such as narrative-driven indie games like *Disco Elysium*, and experimental puzzle games like *Gorogoa*.

In the future, and with technology like AI and social platforms progressing, we can look forward to more dynamic gameplay, more cross-platform experiences, and better tools to empower both game creators and players/streamers.



TIP

You can learn a lot from the history of games and game development. I like to work on the beginning bits of a game and get it off the ground, so understanding how other teams started and what games looked like in the early part of their life has helped me set expectations for development and scope. Games like *World of Warcraft* (WoW) may be huge now, but when they first launched, they all had design problems (like loot that spawns on a world basis instead of on a player-to-player basis). The trick is that these problems aren't game blocking enough to keep players from progressing and enjoying the core of the gameplay.

## Leaning Into the Fun

Plenty of frameworks are available for deciding what goes into a piece of software or a game, but I'm about to say the quiet part out loud: Frameworks aren't always the best way to create a truly fun game. I think the reason that many hit indie games these days were made by smaller or solo teams is that smaller teams have less red tape, which fosters creativity. A lot of the biggest hits were initially made by small teams. *Take Grand Theft Auto* for example; the initial team that worked on the first playable version was made up of 11ish people.

Instead of designing with strict rules or setting up red tape for yourself, making fun the priority should be your guiding principle when deciding which features to include. For every feature or concept you're considering, ask yourself:

- » **Does it fit the game?** Does it align with the theme or the core principles you've established? If not, shelve that feature or concept for later use (we call that "throwing it in the backlog" in the industry).
- » **Is the team (or are you) excited about it?** If the energy and excitement are there, it's likely worth exploring. Developers who are excited about something will put their heart into the design and implementation.

I almost added, "Do we have the skills to make it?" But here's the thing — many great games were made by people who didn't have the skills when they started.

They learned as they went, driven by their excitement. Sometimes, the best features come from diving into the unknown because you're genuinely thrilled about making them happen.



For every milestone I hit in my designs, I like to take a step back and assess where I can trim extra features to focus the game's design and what the most fun part of the game is and why.

## Preparing to Think Systematically

A game designer is holding a lot of interconnected information in their mind. They need an understanding of how the game's systems (the rules, data, and calculations that make the game tick) work on their own and how they interact with each other. If you want to know more about game systems, check out Chapter 7. Every mechanic, feature, and interaction is part of a web of cause and effect, connected by feedback loops that shape the player's experience.

Here's what a game designer should think about when making a game (learn more about what a game designer does in Chapter 4):

- » The core loop and how all the other systems on top of it feed into it to make it rewarding to repeat over and over again
- » How systems can be broken down into smaller, more manageable systems
- » How variables impact the systems so that the systems are fair, challenging, and enjoyable
- » Designing systems for emergent gameplay and unexpected behaviors
- » Creating feedback for player's actions using visual, auditory, or gameplay cues to let players know how their actions impact the gameplay

Systems that you may end up having to design may include

- » Balancing resources so that players don't have a million old boots and not enough wood for what they want to build
- » A procedural system to spawn a new creature with a unique body, tail, and color every time
- » A progression system to level up a character as they complete quests

# Diving Into the Game Designer's Mind

As a game designer, I'm often thinking about how to make patterns apparent in gameplay to teach players what to look out for in my games. I'm also thinking about how to lean into the theme and feeling of the game and how to make sure I think of features I have the skills to create. These are the top things that shape the way I design.

To get a sense of this in a practical way, in a branching narrative I published called *Chief Emoji Officer* (see Figure 1-1), my team and I wrote a big story in which players can make choices and explore the outcomes. In the game, players communicate using only emojis to navigate office politics, climb the corporate ladder, and make decisions that shape their career. But this wasn't enough for us and this game, and we needed to think about how to create a core loop and really drive home the satirical mood of the experience.



**FIGURE 1-1:**  
A screenshot of  
*Chief Emoji Officer*  
(the desktop  
version  
on Steam).

So, we broke up this story into chapters, which helped us do the following:

- » **Create a pattern for players.** There was a summary and evaluation at the end of every chapter of the choices, kind of like a score.
- » **Increase the parody.** We used a system that was like an employee evaluation, aligning with our corporate satire theme.

- » **Create a sense of progression.** As players moved from chapter to chapter, they discovered more unlockable content.
- » **Write the story more quickly.** We didn't risk an overwrite of each other's work (we worked on separate chapters), resulting in less time to finish a chapter.

Game designers are thinking about everything from the feel of the tap of a button to how a story unfolds to reveal a big twist. My favorite game designers have been analytical, scrappy, and ready to build something to test their ideas.

They're often thinking about

- » **How to prototype an idea to experiment and try it out quickly.** This may mean that you sketch something on paper, make a spreadsheet to test some game data, or even build something in a game engine. Learn more about prototyping in Chapter 4.
- » **How players will perceive something and how it'll make them feel.** For example, will a bright red button make players feel urgency? Does the sound effect for collecting an item feel satisfying? Chapter 10 covers creating an engaging user experience.
- » **How to improve features and systems based on feedback from playtesting.** If players consistently get lost in a level, designers may tweak the environment to add better directions. Learn more about collecting and reviewing feedback in Chapter 12.
- » **How to use data to inform decision making.** If analytics show that most players quit at Level 5, for example, designers may analyze the difficulty curve and adjust enemy strength, add a checkpoint, or improve tutorial clarity. Chapter 12 further discusses analyzing data and why it's important to decision making.



REMEMBER

It's important as a game designer to be creative while also being practical. Have big ideas but be ready to trim them down to something more realistic and feasible. There may possibly be a compromise to that online-multiplayer game you want to make as your first game and something that you'll be able to complete in a few months.

## Constructing Your First Game

I'm excited for you to paw through this book and learn some theory and lots of practical knowledge on how to make games. This book may expose you to some resources you didn't know about before and start you in a direction toward whatever specific niche of game you're interested in making.

This book focuses on smaller teams and solo developers who want to make their first game. I include information from larger teams that I've been on, but I had the most fun and got to experiment the most on a team of two, Bodeville, an independent game studio I cofounded with a friend.

Chapter 3 provides an overview of the game design process, and Chapter 14 takes a deeper dive into that process. If you're at the very beginning of your game-making journey, however, you can take these steps to get started:

### 1. Do your research.

Check out similar games in the genre you're looking to make. Throughout the book, I often recommend that you research other games, but I dive into it in some detail in Chapter 4, which covers what you need to know to become a game designer.

### 2. Make a prototype.

This can be in a game engine like Unity, in a spreadsheet, or maybe even using paper on a table. Having a prototype allows you to fully explore the viability of your concepts. Chapter 5 covers tools and software; Chapter 13 covers creating and refining your concept.

### 3. Playtest it with your friends or peers.

Testing it early will get you quick feedback to let you know what's working and what can be improved. Chapter 12 has details about this very important part of the process.

Maybe you have already started your game design journey and are looking for insight for whatever point you're at. If one of the following points describes you, then check out the corresponding chapter to help you with that phase:

- » **You have an idea for a game that you're really excited about and want to flesh out the story:** Story and characters are important! Players will make an emotional connection to your game if you have a great story; check out Chapter 8 for details about developing a story for your game that delivers a unique experience for players.
- » **You've already started developing your concept in a game engine and you're almost finished but hitting some roadblocks.** Maybe you're struggling with scope, polish, or final tweaks. Chapter 15 covers strategies for getting to the finish line, prioritizing features, and managing your development process efficiently.
- » **You have a game almost finished but need tips on publishing and marketing.** Once the game is built, you need people to play it! Chapter 16

dives into marketing strategies, including how to promote your game and build an audience. For details on publishing options, like self-publishing or working with a publisher, check out Chapter 17.

» **You have a game concept but don't know what tools to use to get started.** Choosing the right tools can be overwhelming, but Chapter 5 breaks down different game engines, design tools, and software options to help you get started. The most important thing is just to get started!



REMEMBER

Have an open mind! Some things may not turn out like you expected them to, and they may turn out even better if you have an open mind to feedback and iteration.



TIP

When in doubt, make a prototype!