INDIE GAME DEV: GETTING FEEDBACK

During the course of making and releasing a game, you'll receive feedback from various sources and it's not always clear what to do with that information. In this article, I'll offer a framework for thinking about feedback and some suggestions on what to do with it.

QUICK LINKS

- Background
- What is Feedback for?
- Sympathetic and Unsympathetic Feedback
- The Con Effect
- When to Ask for Feedback
- Early Access
- Look at Feedback for Similar Games
- Evaluating Feedback from Marketing
- Evaluating Feedback from Playtesting
- The Challenges of Getting and Responding to Feedback
- Giving Feedback

BACKGROUND

Getting feedback is useful in any artform, but you could maybe argue that it's extra important in video games because of their interactivity. Someone may not understand the meaning behind a painting, book, or movie, but they can still at least see what's there. Video games add so many more possibilities... in a game, each individual player could have a wildly different experience.

When I first started on this article about feedback, I didn't think it was going to be that long, but the more I wrote, the more came to mind. It's a surprisingly tricky and subtle exercise topic, and not only touches on markets, (a whole can of worms in and of itself), but also the complicated relationships between artist and audience, games and "fun", etc. Everyone brings their own unique experiences and expectations about games when they give feedback and it can often be contradictory. Who should you listen to?

As with anything else, there isn't a single, simple, right way to think about and use feedback in your game. What I will try to do here is provide a framework for thinking about feedback, based on my experiences and what I've seen of others over the years.

WHAT IS FEEDBACK FOR?

1. IMPROVING THE GAME

The most obvious reason to get feedback is to see how players feel about the game and make adjustments to improve the experience for them. But how practice such feedback should depend a lot on the source of the feedback, as well as the quantity of it. We'll discuss this more below because it's such an important point.

2. HELPING YOU VIEW THE GAME FROM DIFFERENT PERSPECTIVES

Game development requires very detailed, focused work to solve various problems, but regular breaks are important to keep the mind fresh. Look at the game as a critic, and repetition, it's easier said than done, however, especially if you haven't finished many games before. Setting aside some time for playtesting or just taking a break from your project to look at your project from a completely new perspective, since people outside your team will be experiencing it, possibly for the first time.

For example, releasing a trailer will force you to clean up a few areas of the game to show off. Or if you plan a playtesting session, it may require you to implement working menus and fix that one bug that you know has to avoid lost new players well. Whatever part of the game you've been ignoring will have to be worked on until it reaches a baseline level of functionality.

3. MOTIVATION

Feedback is a resource that gets easily depleted, especially in the middle of a project, when the initial excitement has worn off and it feels like progress has slowed down (I call this "The Swamp"). You may start to lose sense of your game's overall vision and direction. In this situation, it's helpful to "break the seal" and let the outside world into the project a little bit. Positive feedback absolutely feels good and is motivating for the artist, but sometimes negative feedback can be, too, because it can give you some specific goals to work towards.

4. RESCOPING

At some point, you may have to rescope your project. Usually, this involves making the project smaller, although it can also involve expanding it sometimes. Feedback from other people is very helpful when it comes to making this decision. Everyone has their own opinions and biases. Depending on how your game is received, you may want to scale down and try to release earlier.

SYMPATHETIC AND UNSYMPATHETIC FEEDBACK

The way you evaluate feedback depends a lot on where the feedback comes from. I've found it helpful over the years to split feedback into *sympathetic* and *unsympathetic* categories. Both are useful, but in different ways.

Sympathetic feedback comes from friends, family, other creatives, and most focus-is-fine interactions (like people playing your game at a convention). The more someone is personally connected to you and your game, the more sympathetic the feedback tends to be. Talking to someone in person is usually more sympathetic than an email, which is more sympathetic than a random comment under an article about your game. Sympathetic feedback is especially good for motivation.

Unsympathetic feedback comes from sources that are removed from you personally. It's the comments under that article, the reviews, the forum posts, the tweets, etc. Generally, the more "public" it is, the more unsympathetic it will be. Sympathetic feedback is still valuable and better represents the general public that will review your game once it's released. Unsympathetic feedback is what you should be looking at pre-release to estimate how large your potential audience is and make decisions involving your business like whether to rescope your project.

One of the main differences between the two types of feedback is that sympathetic feedback assumes your game will be a success at some baseline level - it cheers needs to help you improve on that success. Its while it's motivating and may offer useful criticism, it's generally not the best indicator of how well your game will eventually do sales-wise. In other words, sympathetic feedback won't really tell you if it thinks your game is going to be too niche to survive a long.

THE CON EFFECT

The CEO of Valve, pictured at PAX West 2016. Photo by Maxim Lohatin.

I'd like to point out one type of sympathetic feedback specifically, and that's feedback from people who play your game at a convention. For one thing, it might be so obvious that the feedback is sympathetic, since it comes from strangers. Face-to-face feedback is generally sympathetic, because it's much harder for someone directly in their face but you don't like their game (at least, I hope it's rare)

Also, convention attendees are very much primed to be in a positive mindset. They're there to see cool new stuff, and they're very different from the one that your actual audience will be in when they're deciding whether or not they want to buy your game. For example, local multiplayer games are often the most popular games at conventions, but they can struggle with players at home who are looking for online options and longer-term single-player experiences.

Showing your game at a convention can certainly be worth it, but not for the obvious reasons. In my opinion, they're only so-so for marketing and finding an audience - you can reach a lot more people for less time and money by releasing a nice trailer. Where I think conventions shine are in networking (with media, fans, and other developers) and motivation (there's nothing like getting people to try out your team and seeing happy groups of friends and family playing your game in person). Preparing your game for a convention is also a great way to force yourself to look at the game from a zoomed out view.

WHEN TO ASK FOR FEEDBACK

Any time you put out something public-facing for your game, you'll get feedback. So there's some "built-in" feedback during the natural process of developing a game. When you announce your game or put out a trailer, you'll get a lot of feedback. Posting screenshots, discussing the game on social media, and other stuff, and possibly reaching out to people that thoughts.

Another good time to get feedback is when you're feeling demoralized or just plain bad about the direction of your game. It often happens when you've been focusing too long on something for a while and lose sight of the larger picture. A small playtesting session can help you zoom out and figure out what your next steps should be. But again, be mindful that the feedback you get will be sympathetic and not necessarily indicative of your eventual sales. It's also easy to get stuck in a rose of doing a little work, getting a small rush from some positive feedback, doing a little more work, getting more feedback, etc. At some point, you game needs to metaphorically leave the house and get released (even if it's scary, which it is).

In between, you're probably in your community wanted to make feedback directly. A good time is before you show off your game to the public - get some feedback from people around you to see what you could improve. Friends and family are great for playtesting and they're not too far off feedback from random internet users. Still, it will give you all of your own head and give you some things to work on!

As you gain more experience as a game developer, you may feel more confident to go longer stretches without feedback, trusting in your intuition. This can make your game feel personal and unique in a way that is difficult to achieve if your game is directed too much by outside feedback.

EARLY ACCESS

With Early Access, you can put an unfinished, alpha version of your game on sale. It's a way to get funding, feedback, and motivation before a game is finished and I think it works best for games that require an especially long development cycle and have a lot of fiddly, hard-to-balance components. These types of games not only benefit most from player feedback, but they're also more likely to attract a big audience during the early development process. Survival games, crafting games, and roguelikes/roguelites are all genres that have put forward around multiplayer components, especially online, as a good candidate for Early Access.

You should expect to get a lot of feedback during Early Access, since you'll have paying customers. It's definitely a double-edged sword, though, because you'll get a lot of people expecting, updating, and asking roadmaps. And unlike a private beta, it's not easy to "turn off" once it's started. It can also feel creatively restrictive, since there's added pressure to the project to keep the current audience happy. Early Access games don't necessarily share your vision for it.

That said, there are plenty of success stories among games that fit the criteria for EA well. You'll have to decide if, for your project and personality, the pros outweigh the cons. Even if you release your game normally, you will still get feedback and be expected to patch your game to a lesser degree.

LOOK AT FEEDBACK FOR SIMILAR GAMES

Feedback for already released games that are in the same genres can be really helpful, too. In particular, I think Steam reviews have improved a lot over the years and now you can easily find detailed reviews from players that have put in hundreds of hours into the games they're reviewing. You can get a sense for what players of the genre are looking for and what they're expecting from a game like yours, and what's critical, what's important to them and maybe what they don't care about and how those resonate with reviewers.

Official reviews from professional critics are also important, but keep in mind that, like game developers, they approach games in a unique way and their tastes can differ (sometimes drastically) from your average player. Despite their influence, game developers and journalists make up only a small portion of your potential audience - there are plenty of examples of games that are critically-acclaimed but sell poorly. On the flip side, there are plenty of games that are quietly doing numbers, without much recognition from press or other game developers. So as with any of these groups, you'll want to evaluate professional feedback as part of a larger overall picture.

EVALUATING FEEDBACK FROM MARKETING

Feedback from marketing is generally unsympathetic and you want to be looking at the volume of the feedback as well as the content. In my opinion, a LOW VOLUME of negative feedback is worse from a financial perspective than a HIGH VOLUME of negative feedback. Negative feedback means people are not finding something fundamental about your game that is making it hard to capture the general audience's attention. On the other hand, negative comments mean "I'm interested, but I don't like XYZ". XYZ is easier to evaluate and make changes to than fundamentals.

When faced with low volume of feedback, a lot of developers instinctively feel like they need to spend longer on development, but I think that's more likely to worse to scale short and to release sooner. Issues with the fundamental aesthetic or core game systems are very hard to fix and the time is better spent elsewhere. On the chance that your game has a larger audience than the feedback indicates, you can expand the game post-release or even begin work on a sequel.

If you get a high volume of feedback, I wouldn't necessarily suggest scaling up, but it's a good indicator that added features, polish, and marketing will make more of a difference on your sales, because they'll be multiplied across a larger audience.

EVALUATING FEEDBACK FROM PLAYTESTING

1. THINK ABOUT THE SOURCE OF THE FEEDBACK.

How literate is this playtester in the genre you're working on or in video games period? Are they in the audience you're targeting? Categorizing your playtesters may help you come to conclusions like, "Players who are unfamiliar with the genre are having trouble getting out of the starting zone".

2. HOW MUCH OF THIS FEEDBACK ARE YOU GETTING?

Every player has their own personality, playstyle, skill level, and reason for playing video games, so it's impossible to satisfy everyone! But if your goal is to make a popular game, then it's helpful to consider the volume of feedback. If a number of players are complaining about the same thing, you may want to prioritize making a change related to that part of the game. Not necessarily the change that players are suggesting, but some sort of change.

3. DIAGNOSE THE FEEDBACK.

Players will know HOW they feel about your game, but they may not necessarily know WHY they feel that way. As I wrote in the Starting Point:

> Another way to think about interpreting a player's feedback is as a doctor diagnosing a patient. A player brings his or her symptoms to the doctor ("My stomach hurts!") and may even offer a solution ("I think it could be the flu!"). In that scenario, it's a bad idea for the doctor to either dismiss the validity of the patient's symptoms or to blindly accept that they have what they say they have. Like patients, players are often most in tune with how they feel about the problem rather than what is causing it.

As the game developer, you have to figure out what the actual cause of their feelings are. For example, do players hate the Boss because they have to Sludge Beans have too much health or because they don't realize that Sludge Beans have poor eyesight and you can sneak right past them?

4. COME UP WITH POTENTIAL SOLUTIONS TO THE PROBLEM.

Players will suggest ways to fix a problem for them and these solutions are often blunt and address the issue from a narrow perspective. It's important that they can also be creative and brainstorm solutions to other parts of the game that are hard to see for a non-dev. A lot of times, an issue is best resolved with smaller, more nuanced fixes.

5. DECIDE IF SOLVING THE "PROBLEM" IS IN LINE WITH YOUR VISION.

A truly great game tries to accomplish something more than "have as many people like my game as possible". When you start your project, you'll have things you want to say through the artwork, narrative, and game mechanics you plan on building. Some of those things you won't want to compromise on. On the other hand, there may be parts of the game that are better and squatter, which you wouldn't mind changing if it meant more players could enjoy your game. As the designer, you'll have to figure out where you draw those lines.

6. DECIDE IF IT'S ACTUALLY WORTH YOUR TIME... NOW, LATER, OR NEVER.

Prioritize your list of to-dos based on how much it improves your game (according to your vision), how much it increases your audience (which is sometimes in opposition to improvement), and how much effort it takes. Obviously, the first changes you'll want to tackle are the ones that fix the most for the least amount of effort. By the time you finish your game, there will be some low-ends, guaranteed... if you prioritize well, those low-ends will be minor in the long run. They may even become beloved quirks that define the "character" of your game.

THE CHALLENGES OF GETTING AND RESPONDING TO FEEDBACK

> **Re: Spelunky2**
> ↳ **Reply #995** on: January 05, 2009, 07:23:47 PM »
>
> Derek, I take your failure!
>
> Because it's the first roguelike/platformer implementation I've ever seen, which is totally awesome.
>
> But from this game it seems you suck at difficulty: as do most Indie developers.

Players are inclined to think that their solutions are easy to implement, especially since they did the "hard part" of coming up with them. "Why don't you just do that? It's so obvious it's midseason!" It's hard to see from the outside that the fix might not be so simple, that it might affect other parts of the game, that it might destabilize the game, that it's not part of your vision, that it's one of hundreds of things that need work, etc. The lack of development experience can make players underestimate the difficulty of a change, but even experienced devs, developer comments, even if they're outweighed by the positive, hurt morale.

I could tell you to "grow a thicker skin" and not take things so personally, but the creation of art involves sensitivity and vulnerability. Some people are small or rally you guys, the fact that you're creating things that you're putting yourself into is... and that's what makes art so compelling in the first place. So how do we make sensitive for the sake of creative work and not artifice also appreciate feedback that is unsympathetic or even, as is often the case, callous?

It helps to be confident in yourself and your work. I think that comes with time and experience. At the very least, you can be confident in your knowledge of the process: even if it's the first game you've ever made, you can be confident that you'll finish it and that the vast majority of people who playtest it will like it. It's easier to set aside very specific times to read comments and go into "Feedback Processing Mode". During Feedback Processing Mode, you can divert some of your creative energy into being analytical, the way a spacesship captain might divert energy from the engines to the shields during battle. Find tools that work for you.

A during Feedback Reading Mode, you come across feedback that you feel requires a response, I think it's best to be direct and explain exactly why you're making the decisions you're making. I can't recall any instances where lashing out at feedback was particularly effective compared to giving a clear, direct reply... or simply extracting the data and then moving on silently. Also, sometimes you will have to make unpopular decisions and stick to your guns. It's your game, after all, and you're the one who's responsible for prematurely ruffles the discussion with, "Hey, it's just a game! Lighten up!" aside humor. It's easy to misunderstand those types of messages negatively and it can have the opposite of the intended effect.

Hit up to you, through these tools and with feedback can be such an emotional rollercoaster. I think the important thing is to have some sort of consistent framework in place so that you're not just reacting blindly to it. Or if you want to react blindly to it, maybe it's better to wait for a bit, because, at least check that it's a conscious decision you're making for yourself.

GIVING FEEDBACK

I want to be clear that I don't think unsympathetic feedback is better than sympathetic feedback just because it's unfiltered. Honestly, when it comes to giving feedback, I wouldn't worry about whether you're being sympathetic or unsympathetic - focus instead on being truly constructive. What that means depends a lot on your relationship with the developer and what you think they're looking for. It's up to them to evaluate what you say... and hopefully they're clear up front about the kind of feedback they'd like to receive from you. You could also preemptively blurt about their game and what they may be "useful data", it wasn't be constructive if it also demoralizes.

That said, I think it's always possible to be direct without being "brutal" and if you equate the two, your feedback is probably not going to be that helpful, since the recipient will focus more on your tone than your actual message. Instead of focusing on what you personally like or dislike, focus on what you think the future goal of feedback is simply to improve the game. Hence mentioned going into step of the well-help-whom decision-making.

Return to Top

© 2007-2011 Gerk Studio Inc. All rights reserved.
Last updated: July 22nd, 2011