

I'm gonna be talking about user interface design in video games, which is kind of a strange beast because the games industry is – I don't know, it's a weird place. It can be dysfunctional. It can be kinda insular sometimes. But it's also full of lots of great opportunities.

And really, when I got into the position – this was back – the first game I – one of the first games I worked on was *Guitar Hero*, and I was kinda told I was coming in as, like, a graphic designer. And what you realize right away is, when you're working on a small team that has, like, tight deadlines, you end up taking on a lot of other responsibility, and you kinda just have to adapt and kinda figure out ways to rise to the challenge. So my role became much more kind of involved in, like, how do I communicate with the player? How does the player interact with the game?

And so yeah, I'm gonna talk about that. But first I just wanted to look at sorta the history of video game hardware, because this is like a place where I have almost, like, no control over stuff. You're making a game for a certain console, so you have to kind of accept the conventions and the input methods and the controllers that it has.

And if you kinda look where games started, like sorta the arcade cabinet was sorta the precursor to home consoles, and these were purpose-built for one game. They had well-labeled buttons – punch, kick. They did exactly what they said they did. And the instructions were right there in front of you, and it's in a social context, so you can watch someone else playing and sort of understand how to play it yourself.

And when that became, like, the Atari 2600, it was kinda designed almost more like a toaster or a radio, and all the options, all the switches were right there on the machine. There wasn't really a thought that a game might have an options screen or something, or sorta different things within the game. It was just – it actually even has – I think this is funny – has a hardware difficulty switch, so you can set easy, medium, or hard right there on the console. And that's something that we're probably never gonna see again on a modern console.

But yeah, so you follow that evolution, and you get to, like, the NES controller. This is where I started out when I was a kid, and probably a lot of people here did. And it's simple. It's got five buttons. Most games you can figure out through trial and error, like *Super Mario Brothers*. So you can basically pick up the controller and start pressing buttons and see what happens, and kinda like, "Okay, oops, I died."

*[Audience laughter]*

And so yeah, that's pretty easy to grasp. You don't need a lot of handholding. You don't need to explain too much what's going on there.

But then you contrast that with now – the Xbox 360 controller, compared to the Nintendo controller, has a gazillion buttons, and I'm not even showing you the buttons on top. There's like four more up there. So there's a lot more – even a simple game has to use

that controller, and so you kinda have to take that into consideration when you're kinda trying to explain how a game works to a new player.

And I saw this article recently in *The New Yorker*, where someone who had never played video games decided to try out the Xbox 360 and play a lot of current games. This is, like, an author and contributor. And so here is his experience using the controller: "To begin with, you must master the controller. On the Xbox 360 controller, which looks like a catamaran, there are 17 possible points of contact. You must press or nudge or wobble these various buttons singly or in combination, performing tiny feats of exactitude that are different for each game. It's a little like playing 'Blue Rondo à la Turk' on the clarinet, then switching to the tenor sax, then the oboe, then back to the clarinet."

And I mean, it's a good point. It's so easy to forget that that crazy controller isn't something that's super-obvious or super-easy for everyone. People that are kind of already established gamers have kinda gone through the evolution of the controller and had buttons introduced to them incrementally over time.

So kinda what we do, a common way of dealing with sort of this complexity now is to sort of tutorialize games early on, to kind of teach players, as they're playing the game, how it's played and what buttons to press. And this can be – I mean, this is kind of a – it's a common solution. People like things that they've seen in other games, and they're comfortable with putting it in their own.

So I wanted to kinda – I didn't wanna pick on any current games, so I decided to go back to *Super Mario Brothers*, and let's see what happens if we kinda tried to tutorialize it like it's coming out now. Let's say it doesn't exist. How to play it isn't burned into, like, our collective unconscious. Let's say we have to make it, like, with the Xbox controller. How do we do that? Oh, well, already, there's instructions on the screen immediately. Oh, and it's stopping to tell you what's going on here and what button to press. Oh, yeah, you need to kinda jump like that. And yeah, still dead anyway. And then, oh, you died, and that's what that's about.

*[Audience laughter]*

So yeah, and my job is kind of about that balance, like how much do we tell the player without annoying them? We don't wanna negatively impact the experience. And if possible, we wanna positively impact the experience.

So I wanted to talk – well, first, before I get into it, I wanna say that every studio is different. The one thing that's consistent about game studios is just their inconsistency. So I'm gonna be talking from my experience. I'm not promising you that this is super-universal, although I think it is to some degree.

And this word I wanted to share first of all. When I was working at Harmonix on *Guitar Hero*, the art director, Ryan Lesser, used the word "the metagame" a lot, which means –

kind of in a general sense, it's the actual menus themselves. It's just what you have to navigate through to get to the game.

But it also kind of has another meaning, which is like, it's sort of the state of mind you want the player to be in. It's like how you want them to feel about the game and kind of if there's – if you look at *Guitar Hero*, there's no story there. It's a very simple game, but there's this implied narrative that you're, like, outside a concert and you're looking at these cool rock posters, and it kinda just like puts you in the moment.

And in terms of how user experience is kind of considered, it's definitely not as formally considered as I think a lot of people here might be accustomed to. But here, to steal another video game symbol, here's the Triforce of user experience in game development. And these are kind of the people that are really thinking about how someone is experiencing the game.

So you've got – so I'd be the user interface designer, kind of in the middle there, and I'm sort of drawing influence in two different ways, from the game designer on one side – and this can be more than one person, more than one designer – and sort of the art or creative lead on the other side. And usually any kind of outside feedback that I have, that comes from the play tests, or if we've gotten people to come in and play the game, that usually comes through one of these people to meet usually the game designer.

And here's kinda how they influence it in the general sense. The game designer is all about usability. They want the game to be simple and clear and effective. They don't want people getting bogged down and stuff, or confused by the menus or anything like that. It should be easily explained. And kinda the art or creative lead, they want the personality of the game expressed through the UI. They want it to really match the feel – they want it to be credible. And I kinda need to take all this in and kinda balance it and kinda try to add my own sort of ideas to it.

And so it becomes sort of this thing of usability versus experience, because in a way, games don't have to be super-utilitarian. People are playing them to have a good time. They're there for the experience. So if the UI can contribute to that experience in a way, it's a good thing. And I wanted to share kind of a recent example of that.

So I don't know if a lot of people here played *Brutal Legend*, but this was the last game I worked on at Double Fine, and it was my first game there. And so this is, like, some concept art from it. It was already in development before I started there, and I knew it was, like, this really just completely unique and really crazy-looking game from Tim Schafer. And it just had this very – it had this reputation as being a very different, unique sorta game.

And so I kind of wanted to find a way to represent that, so when I was kinda tasked with creating the front end, or the main menu for the game, I pitched this idea to Tim Schafer, who's the creative director: Since so much of the artwork is inspired by sort of these gatefold albums from the '70s, what if we just made an actual gatefold record album and

just, like, took all this sort of influence and pushed it into that, and then we have this actual kind of physical album that we throw in front of the player, and that's what they navigate through. Like, that's the first thing they see when they turn on the game. And he was into it, and that's what we ended up doing.

So this is the menu. Actually, I'll just let this play. So basically what happens is you're kinda – Jack Black is talking to you as the player, takes you into this record store, and grabs this record outta the back and puts it down. And right away the player – I mean, it doesn't really look like a menu, but you see "Press Start." You press it, it opens, and you kinda realize you can control this sort of unique interface in front of you. And it's sort of intuitive because people know how to – a lot of people, at least, have looked through a vinyl album, stared at the artwork, taken the record out, just examining it, especially the kind of old – these gatefold albums, which have just crazy art all over them.

And that became the menu, and what you're seeing here is the final version of it. And what's funny is – I was really happy with the way it came together in the end. It was kind of like an ordeal to make it happen, but it came together, and it was awesome. And this was, like, two months away from shipping the game, and I loved it, but some people didn't. Some people were worried about this. In fact, our publisher was really worried about it because they were concerned that because you couldn't see – I mean, obviously, the usability isn't great. You can't see all the options at once, and they were worried people are not gonna find multiplayer; people are not gonna be able to make sense of this; this doesn't look right.

So then we had to kinda come up with all these sort of solutions to try to solve the problem, which may not have actually been a problem. In my mind, it wasn't a problem, but sort of all these ideas were thrown around of ways of just putting text on the screen, and none of them felt right. They all kinda felt like they were detracting from this kinda cool experience.

And kinda my first instinct when I was kind of thinking – I was being told "We have to solve this, what do we do?" Well, maybe we can tutorialize it. Maybe we can explain it to the player. So I kinda created this, and I got a response from Tim saying, "Man, that's like a big old apology for our awesome album."

So after that, I decided to just kinda back off, and we couldn't find a solution that everyone was happy with, so it ended up shipping without text all over the screen. And people loved it. Gamers really responded to it, like, all over the Internet when the demo came out. It felt different, but it felt right for the game. It really kind of emphasized what people wanted to see from the game, and it was unexpected, and I think that's one of the things that people really enjoyed about it.

So yeah, and another thing it did was it even created this discussion where people started talking about, like, other interesting UIs and other interesting menus they had seen in games, and it was cool because you don't really see that that often. Gamers don't usually talk about UI unless it's really, really bad, so it was nice to see that for a change.

So why isn't there more challenging UI design? Why isn't there more weird stuff out there? Or lessons I've learned. Well, there's a lot of challenges. It's tough to get people to buy into your idea, 'cause you really need that kind of support if you wanna do something a little weird. And obviously, an easy, simple UI is always gonna be less controversial. And it's less work, and that can play a big factor.

But people – the usability, like, a very simple, easy-to-use menu is – the advantage of that is obvious during development, 'cause people in the office are gonna be playing the game all the time. So if something is simple and easy to use, they're gonna respond to that. If something's different but has a lot of personality and is kind of, like, unique, some people will love it, but some people will be kinda nervous about it, 'cause it's a little harder to use.

And the reason I mention an easy UI is also less work. It's because there's like an unglamorous part of my job, which is localizing to all these foreign languages. Like, if I put art in a texture, then I have to remake it four times over for, like, Italian and German and French and Spanish. And the tools we use, aside from Photoshop, which is the same everywhere, most of the tools that I use for getting stuff into the games – I've worked in three different studios, and I've had three different tools, so there's always a learning curve involved.

And it's a one-person job. You have a lot of responsibility thrown onto you. But I also actually put that as a good thing. Because it's a one-person job, you can actually make a huge difference, especially on small teams. It's basically just a matter of saying, "Hey, can I do this cool thing?" and hopefully someone else is into it too, and then they'll be like, "Yeah, go ahead."

And you're also responsible for the game's first impression, just the packaging. I know a lot of us here are probably Apple fans, and Apple packages their products really well. And putting a good metagame around a game is kind of the same thing. It's the packaging. It's what gets people excited, and it's what reflects the style of the game.

Yeah, so to wrap it up, I just wanted to talk about the future, because I often talk about how complicated controllers have gotten, but we're actually at an interesting point now, where we're starting to see kind of – instead of each system being this more powerful, more advanced console, we're starting to see that go in different directions now.

Like, you had the Wii, which wasn't super-powerful but had this different control mechanism, and people responded to that. And now there's like – iPhone gaming has become huge, and that was never intended as a gaming platform originally, but now it's bringing a lot of people that might not have been into games, getting them introduced. And Microsoft has Connect coming out soon, where there will be no controller at all, and I think that that's another great way to bring in people that might not be really familiar – are really kind of daunted by that complicated Xbox controller.

So yeah, it's an exciting time to be in UI. There's gonna be a lot more sort of opportunities going forward for people that are interested in getting involved, and it pays to be bold. And that's my talk. Thanks.

*[End of Audio]*