The following content is provided under a Creative Commons license. Your support will help MIT OpenCourseWare continue to offer high quality educational resources for free. To make a donation or view additional materials from hundreds of MIT courses, visit MIT OpenCourseWare at ocw.mit.edu.

PROFESSOR: This is our first lab session for CMS.608 and, for folks, who wasn't here this past Wednesday? One, two--

AUDIENCE: Who wasn't?

PROFESSOR: Who was not here on Wednesday?

[INTERPOSING VOICES]

PROFESSOR: And so, you weren't here. Let's see, we have some blank index cards underneath the--

AUDIENCE: The plastic Tupperware, I think.

[INAUDIBLE]

PROFESSOR: So you can just fill in your name, your email address plus favorite board game or card game. The email address, basically, just gets added to the email list. Did anyone not get an email from Jason yesterday?

AUDIENCE: Yeah, last night.

PROFESSOR: You didn't get an email?

AUDIENCE: I don't think so.

PROFESSOR: OK. So the email basically said that there was a swap

[INTERPOSING VOICES]

PROFESSOR: --of Friday the 17th, and the 20th. So, basically, we're swapping your readings around-- and that's really the only thing, right?

AUDIENCE: Oh, and I fixed the link for Zimmerman's article.

PROFESSOR: And, unfortunately, Eric Zimmerman didn't migrate his website terribly well, so all of his pictures--

[INTERPOSING VOICES]

PROFESSOR: I actually have one of his games, SISSYFIGHT 3000. In fact, this is before SISSYFIGHT 3000, so this is probably 2999, which is going to be the non-digital version, which is the whole point. It is actually a computer game. It is a pretty decent flash multiplayer game, back in the day when flash multiplayer games were not that common. And if you haven't played it before online, after today, especially if you had a chance to play with the card deck, try out the online version. Do a Google search for SISSYFIGHT 3000, and try it online. Hopefully, there are still people playing it, because you need kind of like a minimum of five people to get a good game going.

AUDIENCE: [INAUDIBLE] the one that was taken down.

AUDIENCE: Aw, suck. It was such a great game.

PROFESSOR: Never mind. Basically, what you had was, you got avatar customization in your game. You could create your character, and you could-- you had this really cool retro pixelated look back in the day when retro pixelated stuff was not that popular yet. But the game-play was pretty much exactly the same as what a card game has.

The readings today basically covered a lot of jargon that we're going to be using a lot of that for the rest of the class. For the most part, I'm not going to go through every single one of them because you folks can read. And if I mention something and you don't get it, you can always check back there.

So just remember that the first chapter in Braithwaite is a pretty good glossary of terminology that we're going to keep using. But one of the big things that they ask is, what is a game? And it's one of those-- Does anybody actually want to offer their definition of what a game is?

AUDIENCE: A game is something that involves players and they set up, define rules beforehand, rules that you have to abide by, whether they be, you can't go outside this limit, you have to abide by certain questions and orders. Something that has structure that is meant for one person to go from Point A to Point B.

PROFESSOR: OK. Structure, rules, limits, players. Anything--

AUDIENCE: It's fun.

AUDIENCE: Fun.

AUDIENCE: It accomplishes some goal.

PROFESSOR: Goal.

AUDIENCE: So you always have some goal.

AUDIENCE: You're working towards something.

PROFESSOR: Anything else?

[INTERPOSING VOICES]

AUDIENCE: Or it's an abstract system. Often an abstraction of some real system.

PROFESSOR: An abstract system. So these are actually pretty good benchmarks. Now there are, again, as Brandon describes, there isn't one that's just the canonical answer for everything. It really depends on which theorist you are reading at any given time.

But usually it will involve a significant subset of all of these in the definition. And I like to keep things broad and vague because, for the most part, if you define yourselves too strictly, it might be useful. You're an academic and you need to write papers about it, but if you're a designer and you're just trying to create something fun-- the word I would use is "engaging," because fun means a lot of different things to a lot of different people.

Engaging basically means that this is going to be something that you're going to want to do for a while, that is motivating. It might be terrifying you out of your wits. Some people enjoy that, or would find that experience compelling. I mean they may not call it fun. So I call that engagement.

The problem with using fun and engaging as a metric for defining what a game is, is that that means that a bad game is not a game. A game that fails to achieve fun, that fails to be engaging, is not a game. But I wouldn't go that far, largely because that means a lot of stuff that I created is not a game.

But it should give you the-- there is no magic point where a game becomes --the thing that you're building becomes fun and then suddenly becomes a game. I can't even imagine someone making that argument, that it's not a game until it's fun. But I much rather sort of say, fun and engagement is kind of more of a metric of how well designed your game is, not necessarily whether your game is a game. But that is a goal of a game designer, to try to create an engaging experience, something that people are going to want to do.

I know some game designers who think that is not, in fact, their job. Their job is to maximize revenue, for instance. Especially if you work in social games, there's a lot of discussion now in how business and game design is basically just one thing now. The whole point of game design is, how do we get people to pay a little bit more? Zynga's going to be in town on September 20th, they're giving a talk. That has those two points in it.

[INTERPOSING VOICES]

PROFESSOR: So I always thought of design as something that you're working towards. But all these other things-- a game should have players. Again, I don't want to be too strict on that because I've certainly seen games that play themselves.

Anyone here involved in Battle Code? That game pretty much plays itself, right? I mean, you set it going and then one side wins, but there's no human person at least during the time when the games actually being played. Or you could argue that the whole course of Battle Code-- for those people who don't know it, it's an IP course, I believe, where Course 6 students basically write bits of AI to play a real-time strategy game against some of other team's AI. And the basic idea is that, you win a real-time strategy game, your AI wins, but there's no human actually playing it at a given time. But, of course, there are humans writing the AI, so if you think of the entire class as a game, then there are players, there are people.

I suddenly see people make a case that there are games that are zero player, and I've heard, for example, there's a game out there called Novick, which is largely a game about coming up with rules. And there is a system, which are actually in the rules themselves, about how you change rules. And, of course, those rules themselves can be changed. And that changes the way how you change rules and, by the time you actually play the game, someone's already won. Because somebody in the game has no legal moves. That's the whole game, is basically making it impossible for someone else, or someone else except for you, to win.

So does that game actually have players? Well, yes, it does because you guys are arguing over all the rule changes through the rules of the game. But, in a sense, it had no players because the game never actually got played. The rules are basically stating, this is an unplayable game for everybody except for one person who's playing the game.

Goals are one of those things that I used to be very, very adamant about. Like a game had to have a goal. You have to know what they're going towards, and if you don't have that, then it's a toy or something, it's a dollhouse, it's an experiment. The Sims was the one that I was like railing about when I was an undergraduate. You know-- that's not a game. And, of course, I was working with many colleagues and eventually came across, came over to decide that it doesn't matter whether I think it is a game or not. If it's in the game shelf of Best Buy, it's a game. Because people think it's a game, and if enough people think it's a game, then it might as well be a game. Otherwise-- because that's what people think it's going to be.

So when you come up with your game, I would say that goals are one of those things that helps steer player behavior. So it's one of those tools that you have as a designer to basically say, this is important, this is not. Reaching the end of 100 meters is important for a sprinter. That end is really not interesting for somebody who is running 400 meters or a 50 meter dash. But that gives someone a direction for them to go towards. But there are plenty of games where you have to invent your own goals. Other than Sims, can anyone think of any? Where inventing your own goals is kind of the fun.

AUDIENCE: A lot of NLRPGs.

PROFESSOR: A lot of NLRPGs. They have goals. They have quests. But that's not where a lot of fun is, necessarily.

AUDIENCE: Second Life.

PROFESSOR: Second Life. A lot of virtual worlds.

AUDIENCE: A game like EVE?

PROFESSOR: EVE? Yeah. EVE, you kind of, want to, like you have to set your own goals, because whatever is in the game is only good enough for a single player-- to tell whether a single player whether they're doing well or not. But EVE is not about a single player. EVE is about thousands and thousands of people playing simultaneously. Actually, have people heard of EVE?

All right. Yeah. So, cool. Cool.

AUDIENCE: [INAUDIBLE].

PROFESSOR: Oh, cool. In Iceland?

AUDIENCE: Yeah.

PROFESSOR: Was it like really cheap when you went over there?

AUDIENCE: Yes.

PROFESSOR: Could you have bought the entire studio?

AUDIENCE: No. Unfortunately. But it was really cheap.

PROFESSOR: So they must have been making tons of money over there.

AUDIENCE: Yeah, they're making money in US Dollars and the Euro.

[INTERPOSING VOICES]

PROFESSOR: They probably hire most, yeah, support most of the economy.

[INTERPOSING VOICES]

PROFESSOR: Yeah.

[INTERPOSING VOICES]

AUDIENCE: The capitol is 20,000 people so...

PROFESSOR: So I'll say goals are very important tools and as a result of that you see them in a lot of the games. And for most of the games that you guys are going to be building, you're going to give a very clear goal of this what you want to do-- earn a certain amount of money, prevent your opponent from doing anything else, take over all the land. That sort of thing.

Let's see. Oh, I'll get to the rules last. But, system. Systems are-- I might already have mentioned this in the last class, but games are systems in a way that most MIT engineers would think of systems. They're a bunch of interconnected little modules, all of them with sort of predictable behavior, and then you put them into a big interconnected system, and it is no longer really all that predictable anymore. At least that's the kind of games that we're interested in in this class.

There's also the game-theory games, where the whole point of it is trying to predict what could possibly happen. And those are great thought experiments and there's some really, really interesting math behind all them and some interesting rules of thumb that can come out. But for the most part, we aren't going to go into too much detail about that, largely because I'm not a mathematician or economist, so I don't really have a good sense of game theory. Just be aware that there is this whole field of game theory in economics.

If you're doing a game like an auction system, if you're doing a game where people are doing either a lot of simultaneous moves or the whole point of the game is trying to predict what strategy your opponent's going to do, you might actually want to read up a little bit on game theory, if nothing else, just to give you some vocabulary to talk about it with your teammates, and some tools to be able to think through some of the design problems that you've got. And we do have one session where I'll introduce some of that.

Finally, we are getting to rules. The things that give your game structure and the constraints of the things you can't do. You can't move from here to there without having your foot tied to another player, that sort of constraint. Or a rule being you can't see anything during a certain part of the game.

There are two bits of terminology introduced by Brenda and Ian Schreiber in their book, which is in *Core Mechanic and Core Dynamic.* So before I'm going to go into the core mechanic and core dynamic, I'm going to talk about mechanic and dynamic.

Anyone want to throw out the definition of mechanic?

AUDIENCE: I reckon it's sort of the actions like a player takes during a game like the physical actions he takes.

PROFESSOR: OK. The actions that a player decides to take?

AUDIENCE: Decides to take, yeah.

PROFESSOR: Sorry if this is a little low, I saw Patrick and-- or did those hands just go down? I thought I saw a few more hands.

AUDIENCE: Actions that are sort of designed by the game designer. So it's intended actions.

PROFESSOR: Design is definitely a good point. A game mechanic is definitely something that you as a game designer control. There are-- if a player decides to do something in the game that you did not explicitly think about, and you did not explicitly say, all right this is what players are going to be doing in a game, then it might not actually be a mechanic. It might be something that you're discovering in your play testing that you will turn into a mechanic. But because it's not designed, it's kind of-- it becomes more like a strategy, actually, like an emergent strategy.

Sorry. You were saying, Jeremy?

AUDIENCE: Is it more like a dynamic ?

PROFESSOR: Possibly. There are times where it is. Yeah. Actually let's throw out definitions of dynamic, what do people-- well, Jeremy?

AUDIENCE: It's behavior that emerges from mechanics?

PROFESSOR: It's emergent behavior? What else?

AUDIENCE: Use of mechanics that isn't directly stated in the rules of the game.

PROFESSOR: OK. So unstated. I'm running out of space here. I thought I saw a hand on this side. No? OK.

All right. So, yeah, dynamics are basically interactions of mechanics. It's like you don't have anything that may necessarily explicitly say, because of this one rule and because of this other rule, and because of the things that the players can do in that, a good strategy is to do something else. So let me see whether I can think of an example. Right now, everything that's coming to my head is StarCraft.

AUDIENCE: Has that been the case for the last few weeks?

PROFESSOR: Unfortunately, yes. By the way, if anybody is playing StarCraft, I need testers.

AUDIENCE: StarCraft II or StarCraft I?

PROFESSOR: Two.

AUDIENCE: Ah, yes.

PROFESSOR: I need testers for a math game I'm designing.

AUDIENCE: I play Zerg.

[INTERPOSING VOICES]

PROFESSOR: OK.

AUDIENCE: So there's this role selection mechanic where everybody picks the thing which they want to happen during their turn.

[INTERPOSING VOICES]

AUDIENCE: and mostly everybody chooses to build a particular cross. And the dynamic that emerges from the sequence of play, which goes clockwise is that it's best to optimize your strategy so that you were doing, so you were benefiting from the role of the person to your right, is likely to choose. But you're making the [UNINTELLIGIBLE] of the person you elect is most likely win a ship. So mechanically it's role selection and turn order and then the dynamic is the strategy written by the program.

PROFESSOR: Which gives you some idea what to select as opposed-- the mechanic is you can select. The dynamic is, here are a couple of strategies that you can use in sort of assessing what would best to select. And if all the players start to understand that, then it almost becomes like a second layer of rules almost.

If anyone has played a game like Bridge, for instance, there's actually a ton of things that are assumed in the play which are not actually in the rules of Bridge when it comes to bidding. Like when you call out a bid, it's highly dependent on what you actually have in your card hand. And that's not actually written in the rules anywhere. But the assumption is that there is optimal way to do it. And since there's optimal way to do it, everyone's expected to know those optimal ways.

The example in the book was chess moves, like every single piece of chess has a set of moves that it can perform, right? Pawns go one or two squares, depending where they're starting. The bishops move diagonally. But there are also opening gambits. Opening gambits that are also well known, at least among professional chess players, and the idea being, oh, this person is using this opening, therefore, if I counter it with this response, I actually have a reasonable chance of actually winning this game, and professional chess players know this.

The other thing that comes up in chess, which is not in the book, is there is the concept of threatened squares. If you move, if there is a square where there is no piece on a chess board-- everyone here knows the rules of chess, more or less? So say you've got a bishop-- and this is quite a simple chess board-- and you've got the bishop, and it can move like that, right? So every single square on these lines is threatened. It's something which the bishop could move to. Unless, of course, it's lost by a pawn or something, then these are not going to be threatened by the pawns.

If your knight obviously is no longer straight lines, it becomes a specific or sort of like stochastic pattern of squares that become threatened. But the idea of here's this square now that nobody dares move into until something's done about that bishop. Either something gets moved in its way, or something takes out the bishop or forces the bishop to move by threatening it instead in a way that it cannot respond. So threatened squares is not a rule in chess, but it greatly shapes your decision making and that's the difference between dynamic and mechanic.

There's a lot of strong correspondence between mechanics and rules. But-- and I had a long, long little rant about exactly what the difference is between mechanics and rules. And what I basically-- I know I'm going to get the rant right now, but it's like overlapping venn diagram. There's a very, very large area where it's exactly the same thing. My general rubric for trying to figure out whether something is a mechanic or not is whether this is the thing that changes the state of the game. It is an action. It is something that someone is going to be doing to change the state of what the game is. In chess, that's largely the movement of the piece. If it's ultimate. It could be just physical movement. You are limited on when you can physically move. If you have that frisbee, you're not moving. Well, your feet are not moving. I believe that's a rule of Ultimate.

So the reason why I use that definition is because there are occasionally times when I need to talk about mechanics that players aren't actually doing. The game requires certain pieces to move in a certain way, certain things to happen in the game in such and such a time. For instance, you have these die rolls and-- actually, that's not a good example. Let me think.

Right at the beginning of Settlers of Catan there is the way how you set up the board, and that establishes what a game state is going to be. It changes game state by giving you a game state in the first place. I consider that a mechanic. That is something that's going-- that's not one rule. That's actually a whole collection of rules. And if you're interested in finding out what those rules are, the box is actually in that closet over there.

There are something like five or six different rules on how you set up your board. It's going to be, set up your game state, and I consider that a mechanic. How people do bidding and buying things in Settlers of Catan? That's dynamic. There are no real rules on how much something costs. That's up to the players to work out on your own.

So that's how I sort of consider the difference between dynamic and mechanic. They're really, really useful concepts because really, for the most part, as a designer you have an idea of what kind of dynamics are interesting. And the more games that you play, the more inspiration you get from various sources, your range of different ideas of what kind of dynamics might be interesting is going to expand.

But all you have control on are mechanics, the specific rules that you're giving, note that game mechanics don't necessarily need to be written or described out. This is one thing that I think you and your teammates should figure out when you're talking about mechanics for a given project. Are we just talking about the things we are writing down? Because there are some assumptions that are just given. We are playing a game, and we're not playing some sort of schoolyard game or something, hitting people with solid objects that's disallowed. Not necessarily something you need to write down. Unless you're playing with certain people. You can assume that the game again -- things like how fast people normally do things.

There's a game over here called Falling, which is actually a real-time card game. It goes as fast you can possibly play it, and there is no there is no limit on how fast you can play these cards. But there is a real physical limit on how fast you could physically grab the card out of your hand and put it down. So, actually, it's not even on your hand, your just moving cards on table. So make sure you give that a try today.

Finally, we'll be getting to aesthetics, I think-- next class?

[INTERPOSING VOICES]

PROFESSOR: So, which is another thing which a game designer-- in fact, what I would like game designers to actually spend more time thinking about, which is, what is the feeling you want your player to have? As opposed to, this might be fun, or this might be engaging. And specifically, how is it fun? And how is it engaging? Is it fun as in like, yes, victory! I have crushed all my opponents. Or is it fun as if, well, we had a really good time together and we got to know each other really well. There are games that are designed for both of those.

And what is the aesthetic you're going for? Are you going for something poetic? Are you going for something fiercely competitive? Tournament competitive? Are you going for just a nice little social experiment?

Core mechanics and core dynamics are a tool, again, for designers. The idea of-- there really isn't a huge difference between what's a core mechanic and what's just a mechanic, except inside a designer or design team's discussion. The idea of having a core mechanic is you identify this is what our game's about. Or a core dynamic means this is what our game's about. And a lot of core mechanics in first person shooters is pointing and shooting. I'm going to point at something and I'm going to launch a projectile that way, and that projectile is going to move in some physical properties. Maybe it arcs, maybe it goes straight.

I would say that the core mechanic of Halo isn't actually that. Who has not played Halo or seen someone play Halo? OK. So the idea, core mechanic-- or I think my Bungie calls it the core game play loop-- of Halo is, you shoot at something, you take cover, you throw a grenade at it, you wait for the grenade to go off, and then you shoot at it again until it dies. That's pretty much all you are doing for the entirety of *Halo 1.*

I'm sure *Halo 2, Halo 3* gets complicated, but de-identify that as you actually read the post-mortems and if you read the talks that are given, they say those-- I can't remember if it was 20 seconds or five seconds of game play which is used to describe that. I think that five seconds of game play is what they focused on. And if they could get every little thing in that sequence correct, they can repeat that same thing at infinitum, and you'll still will be having fun. *Halo 1,* everyone.

They're kinda right. But the idea being, if you've got a core mechanic, and you've got one thing in your game that someone's going to be doing over and over and over again, all your players are going to be doing over and over again-- maybe it's bidding because it may be some kind of auction game. In Falling and Pit, it's transferring cards to other people. I'm trying to think of the other games that we got here. In SISSYFIGHT, I would imagine the main core mechanic would be putting down the card and everyone revealing the cards simultaneously, because of you all take simultaneous turns. Make sure that's really, really fun.

And Yahtzee is rolling dice. That action, that thing that you're going to be doing over and over again, has to be at least fairly engaging. Because if you can't get that right, then what happens is that that same movement, that happens 30 to 50 times in your game, just-- the badness just gets multiplied by 30 or 50. And you don't want that. So for core mechanics, I would say the thing that happens in your game the most-- and you want to make sure that that part is as engaging as possible because it gives you a rubric to say, what's not a core mechanic might be, in fact, expendable. You might not actually need it in your game. So if you've got a problem-- this game takes too long to play, this game's too hard to learn, this game's over too fast, it's too random or something-- take a look at all the things that are-- but first make sure that your core mechanics not the cause of the problem.

But if the core mechanics doesn't seem to be causing the problem, that seems to be working, something else in the game is not working. Everything else expendable. And so you can look at everything else in the game, all these other rules that you've got, and say, how does that work with a core mechanic? How does that help? How does that hurt the core mechanic? It just helps your discussion as a team. Or as an individual game designer, it helps you to think about what could we throw out? What could we keep?

Similarly for core dynamic. It's like here's this particular interaction that we really, really want. And SISSYFIGHT is guessing what other people are going to do. I'm trying to think of other examples. And like territorial space acquisition. I guess Go is about taking over territory in space. If that's your core dynamic, you say that's kind of fun, just making sure that I'm in control of space and nobody else can do anything they want to do in a certain part of the board. That's fun. That's what I'm going for; that's my core dynamic. That's what this game is all about.

Then you can look at every single mechanic, even your core mechanics, and say, is it all serving that goal? Now there's a whole bunch of tips right at the end of chapter one of *Challenges for Game Designers* about how to break designer's block.

But I would say, actually, those things are just things you should be doing all the time, not just when you have a problem. But you should be experimenting. Things like, OK, you've got a variable. Draw two cards every turn or something. Multiply it by two. See what it does to your game. Halve it. Draw one card. See what it does to your game. Through a rule-out, and that sort of gets you to the whole iterative design thing, is that you don't actually, because games are systems, these complex little collections of seemingly simple things, you don't actually know how these changes are really going to affect the game. You may have a hunch, but until you actually play it out, you don't actually know.

So the whole point of iterative design in engineering as well as in game design is that you make the change. It has to be a substantial change. And then you actually put it to the test and see what happens. This means, of course, testing, and testing takes a long time and if you're going to be making a game that takes a long time, then your testing gets even longer.

So for the first project that's going to be coming up, assignment one, you want to be aiming for games that take really not very long to play. These games that we got today, I'm not sure about [INAUDIBLE], but I know everything else that we got on that table, is pretty darn fast. Which means if you play an hour with four people who've never played your game before, you get through two or three rounds, four rounds, of them. Each time, you'll be teaching a rule once, sharing and seeing how that changed the game.

If you've got a game that takes an hour to play, and believe me, a lot of the projects that we've gotten from previous semesters take at least an hour to play. You've got four people for an hour and another definition of a single-player game. Because by the time they finish learning your game , half your time's up.

So that's a process that we will like you all to be using in your teams and your assignments, is iterative design process. Right from the beginning-- I wish I had more space-- but we're going to start with brainstorming. That's actually going to be Friday, the 17th, where we'll actually sort of-- for folks who haven't necessarily formally learned brainstorming we're just going to do a quick refresher so that everyone sort of is on the same page.

You learn how to do brainstorming. You come up with a basic idea of a game, and you start to identify things that could be your core dynamics. You think of what sort of core mechanics could address that. Knock something up really, really quickly with the crappiest materials you can get your hands on and start playing.

Since nobody on your team knows, really, what game you're making at that point of time, you can probably test it with your team. As soon as you do it a few times, your team becomes useless as testers because you already know too much about the game and what the game could have been. And you start playing it in ways that a fresh person, who's never seen your game before, wouldn't even consider.

So you actually want to keep looking for people who've never played your game before. So the next step would be to broaden it to other teams in this class. Because everyone's going to be broken up in more or less teams of three or four. That's pretty much, I'm going to take this game that we designed for a team of three or four, and give it to that team of three or four. And then you play their game, and give each other feedback. And that lasts for about one more session, and after that, everyone in the class is useless now.

So you have to look for dormmates, friends, people on the street, just go right ahead and try to get as much feedback as you can. There is some value in replaying, especially later on once your rules are pretty set in that people start developing strategies, still some of them emergent behaviors, things to understanding sort of higher level play of the game, and you do want to test that. But don't let that prevent you from also testing with fresh players. Because in the end, you have to get past that initial hurdle. Anytime someone sees your game, they are going to have to learn your game from scratch at least once. And if they don't even get that far, they're never playing your game a second or third time. So don't optimize for the people who play your game for four weeks, because you really want the people who've only played it once to have a good time as well.

That's not necessarily the case if you're making a huge, well-funded game based on a franchise that's been around forever. If you're doing Halo 3 or Modern Warfare 3 or something like that, then sure you can sort of cater to the fans who've already developed these strategies and skills. But in our case, we're going after a new audience. We're going after someone who's never played anything like what you've played with before.

Cool. Any questions? Feel like we've been soapboxing here for a while. Is there anything?

AUDIENCE: I guess the thing I would add from my experience, especially with iterative design is, a big trick of it is learning to kind of distance yourself emotionally from what you're working on, because half the time you end up chucking things that you really liked. You think, oh, this would be so great, but it's best to just remove it. And it helps to think of it as kind of putting it on the shelf for later, and saying, OK, I'll deal with this later, or I can try this in another game.

And on a related note, I don't know about you, but I found that, when games I'm designing have a problem, it's almost always fixed by removing something as opposed to adding more stuff on it. Unless you're going for something, I think, really complex off to start, I always prefer to try to take something out and simplify it. Especially on short projects like these it almost always works better.

And then, in terms of testing, too, one thing just kind of a tip that I find works, is if you're designing a game that has a lot of different, like, passive victory and this is a little bit higher level, which you'll see when we get there, where if you want a game that has different viable strategies, a really good testing method is to have different people kind of play like a perfect version of that strategy. And you say, I'm going to do this strategy, or I'm always going to do X, and it doesn't matter if I think it's dumb, I'm just going to do it. Because that will really help you find rules that are appropriate, or find something that's overpowered.

AUDIENCE: A good example would be Settlers right. Like if you say, obviously this wouldn't work because, the game's a good game, but like, I'm always going to build roads. You know what I mean? In practice, that would be stupid,but when you're designing a game, you do that kind of strategy, and you just limit yourself to one action, you can really find things that are broken really quickly.

PROFESSOR: Thank you. Much appreciated. Cookies. Good, we have cookies.

Let's see. So I think after this, we're just going to play games.

We do have pre-testing as a topic coming up later in the semester, where I will actually sort of talk about the discipline on how do you do testing. Honestly, you shouldn't wait for that class to necessarily start practicing. It could be as simple as just grabbing a unit that you have never seen before and just trying to get a whole bunch of people to play it. And critiquing it at the same time. It's like, why did I do that? Why did you do that? Why did you do that move?

And there are sort of practices to make sure that when it's your game, you are not sort of telling the players what they should be doing while they are trying to play the game, because that affects their play of the game in a way that biases it. Well, we'll get to that later in the semester. For now just be aware, the more that you're testing, the more you're sort of tweaking your rules in between tests and then getting another chance to test and test and test, the closer you are to getting a game that is actually fun and engaging to play. And that's what we're looking for. That's the process that we expect you folks to suceed in.

When do we ever get to fun and engaging? Who knows? We don't know whether you're going to get there, so we're not grading you based on that. Just remember that. We're not grading you on how engaging your games are. But how well do you stick to the process? Yes. Absolutely.

There was a question in the forums about when do you stop? I didn't get a chance to look too closely at it. What--can you repeat it?

AUDIENCE: It's essentially, it's the question of, say, there is this iterative design process. When do you know when to just sort of get out of that process? But--

AUDIENCE: Due date.

AUDIENCE: Due date. OK. That's always a good reason. I guess, if you didn't have a due date, like how would you know when to shift? Well, let's say you have control over the due date. I guess part of that is, you should always be updating, blah blah, so I guess the question is when do you know when you have something you can put out into the world?

PROFESSOR: So the cynical answer is that, yeah, is that you wait for -- you have to ship, right, or you're not getting any cash. I personally think it would be sooner than you think. As, like, my game's almost ready but I know there's something wrong with it, especially when you're a student. Right now, I'm in this mode with my StarCraft games, which is, this path is cracked. I know there's something wrong with it. I have no idea what it is. Therefore, I'm going to release it. Because until I do that, I have no idea what's wrong with it. I need to get that information.

The nice thing about stuff like computer games is that you can ship updates. It's annoying but you can. But even board games and card games now have rule revisions, rule clarifications that you can release over the net. For the most part, if you're going to go professional, your publisher will tell you. And then the answer really is due date. It's going to the presses on this date. And that's one maxim, which is, game projects are never finished. They're just abandoned. It's just horrible. I can't do anything else about this because I'm out of time. Ship it.

But one thing that you can do to prepare for that-- Let me see. Is there a laser around here? There is a guy who runs a blog called Lost Garden. His name is Dan Cook. He works over in Microsoft. And he talks a lot about game design. Specifically about video game design, but actually his stuff is actually really, really impeccable. And he has this concept that game design-- or iterative game design kind of goes in this kind of cycle, where this list down here is the number of features you've got in the game.

The trick, though, is that what you're really doing is, you are taking-- let me see if this is right. So this is all the good stuff. This is all the stuff that's not so good. And the whole idea of sort of like using iterative design is to allow yourself to the space to come up with a whole bunch of ideas, not knowing whether they're good or bad. But it also gives you a gate to cut out all the bad stuff.

So at frequent periodic intervals, you have a collection of rules, of cards, of designs, of whatever consists your game, that, is pretty much as good as it possibly could have been given the amount of time that you got. And that could ship. So if ship date was here, you had as good a game as you possibly could do it. If ship date turned out to be there, all of a sudden you've still had the opportunity to put out all of the stuff that you knew was bad. So iterative design is not about, let's just keep accreting more and more and more and more features. It's more about, we're going to add some features and then we're going to remove the ones that don't work. And that's really, really important. So at any given time, you are prepared to sort of release it. As for when you actually decide that this is now ready to go out, again I will repeat, sooner day than you think because you need that feedback.

Sometimes you don't know how a market is going to respond until you actually do it, but you can do a soft launch. If you are doing a commercial product, you can do things where, we're going to release it to a few select stores and get that feedback before we go into major print run. Or for 10,000 different copies of the same board game. We're going to 200 copies and see how that does for us. Gives you the opportunity to make a revision, but it's still an official release, because those 200 people think that this is the final packaged product.

That's pretty much how a lot of manufacturing industries actually work, anyway, so you think about any kind of product design, a lot of them have a sort of a trial period.

AUDIENCE: Soft drinks you get a lot.

PROFESSOR: Soft drinks you get a lot. Japanese electronics, and other things. You release it in Tokyo, and then you release it in the rest of Japan. And sometimes it never makes it out of Japan because that initial test-- yeah, it was a good product but not enough people bought it because of these problems. So instead of releasing it into the world, we're going to make version two, and maybe that will be one we release to the world. That's one example.

All right, so we have games. Who has not played Uno? OK, we have more games that--

[INTERPOSING VOICES]

[LAUGHTER] OK. So Jason, you're going to do a Euchre table? So--

AUDIENCE: It sounds like 4 player trick taking game, kind of like bane or current speeds or which?

PROFESSOR: Looks like we have exactly four people-- one, two, three, four, one, two, three-- it's an average of three people per table.

I think most of these games are three to five player, although I'm going to be absolutely sure.

[INTERPOSING VOICES]

PROFESSOR: --so it might

[INTERPOSING VOICES]

PROFESSOR: You guys might actually want to try the round table in the corner?

AUDIENCE: Yeah, it's a [UNINTELLIGIBLE]

AUDIENCE: I really want to play bowling.

[INTERPOSING VOICES]

PROFESSOR: So basically, take a game from the table. I'll bring some out and I'll see --play them. Once you've had a chance to play a few times, feel free to switch. And try to get as many games played as possible during the next couple of hours.