PROFESSOR: So one of the big mantras in any kind of environment is testing. You have to keep testing your game, because no plan meets first contact with the user. You have to be able to see how real people are going to respond to the stuff that you're making as soon as possible, because many of your assumptions going into a project are going to be wrong.

Now, we encourage students to go out and test their games with as many people as possible outside of class. This is crucial for the development process. And there are a couple of assignments where they actually have to show us the results of that testing. And it's pretty much as simple as sitting somebody in front of a computer with the latest version of the game, or maybe with the paper version of the game, giving them a couple of instructions just so that they can get started, and then stepping back, watching, and taking notes. Trying to provide as little guidance as possible, so that they can actually see how a player who knows nothing about the game responds to it for the first time.

However, sometimes it can be difficult to get a lot of testing done outside of class. And students may be a little bit shy approaching strangers, for instance, even other MIT students. So what we do also in class is we get different teams to play each other's games and give the feedback.

The quality of that feedback's a little bit different, because everybody the a room at that point is in the same boat. They are all making games, and all trying to give feedback in the way that a game developer will give another game developer, as opposed to, say, a naive player who just wants to have a fun time playing this game.

We will also be giving feedback as instructors. We give them our experience, both as developers and this instructors. And we can tell them, that's a great idea, but there's no way you're going to get it done within the amount of time left in this semester, because we've seen it. We've seen how these projects tend to go during the class. No one else in the class has gone through this class multiple times.

So this is basically a way to get feedback about the game. And we teach students various methods to get the information they want, and to record it, such as questionnaires, for instance, different observational techniques, different ways to script the process of getting a new user to actually get started. And most importantly, the importance of just doing this a lot.

PROFESSOR: So there are a lot of different ways you can do project management. The one that we teach and we favor at in our course is we look at agile project management techniques. Most specifically, we look at the one called Scrum. And by the nature of agile, agile project management really requires a team to work with itself a lot to understand how the team works together and to understand each of their team members' strengths and weaknesses.

This is exactly not what we give our students a chance to do in the course of the course, unfortunately, because they are working on short projects. Each project tends to have slightly different goals. And they're changing teams frequently.

So what we do do is we give them a set of agile tools. We introduce them to the concept of backlogs, project backlogs, creating very short vision statements to describe what the goal of their project is, but not go into a whole lot of detail. We have them make project backlogs that list all the features they would like to have. We have them take a look at how they break that amount of work down into how much can we get done in a week? And realistically trying to figure out how much they can get done in a week, at which point they realize their project backlog is about nine weeks long. And they can trim down from there before they get started on some of the less important things.

But we show them the tools. We walk them through using the tools. And then we encourage them to use the tools that work best for the group that they're with and the project they're working on. So instead of saying, this is how you do it. Shunk. We say, here are a set of tools. Work with all of them. Use the ones that work best. And try to iterate not just your project, but also the way you are managing your project and the way you're working together with each other.

Because that's the only way to get to a better working team, is to find out what works. Accept that some of your techniques don't work. And try something new, try something better, until you do find what works with that particular group of people.

PROFESSOR: So we're not only talking about the tools and processes, the hard skills when it comes to project management, but we're also looking at iterating on their soft skills, which often comes down to just how do you talk to another person respectfully? We talk about team dynamics.

So teams tend to grow in a particular way. And often, there are shared qualities that all teams have, when it comes from forming from when they're getting their first conflict, and trying to get over their first conflict, to when they're actually performing pretty efficiently, and then when they feel like they're just really kicking butt and really moving forward. We try to help them get through to that final stage. At best, they usually get to right to the point where they really start performing along, really start being efficient in what they're doing.

But what we want to do is have them see the whole process, see how the stages differ, and then, come up with ways and approaches to then get themselves through the process, and being able to talk to their other teammates about, maybe, a problem happened. Maybe somebody didn't check in some code. And it caused two days of work to go away. There are better things to do than just to rant and rail at this other person. So what are the other things they can do to get through that problem?

So we want them to talk about that. We want them to think about that. We talk about it a little bit in class. And we ask them to give us what would they do again? If they had do this again in the future, what would they change? What would they do differently? So we're asking them to do that through their individual write-ups.

We ask for a lot of reflection on the class, on not just the game design process, the game development process, but also this project management process, and the communication between team members, as well.