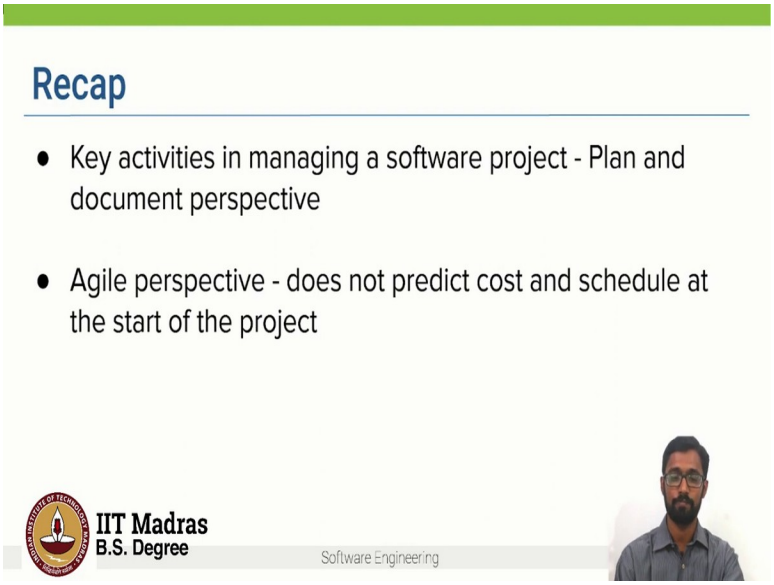
**Software Engineering**
**Professor Sridhar Iyer**
**Department of Compute Science and Engineering**
**Indian Institute of Technology, Bombay**
**Professor Prajish Prasasd**
**Computer Science**
**FLAME University**
**Project Management in Agile**

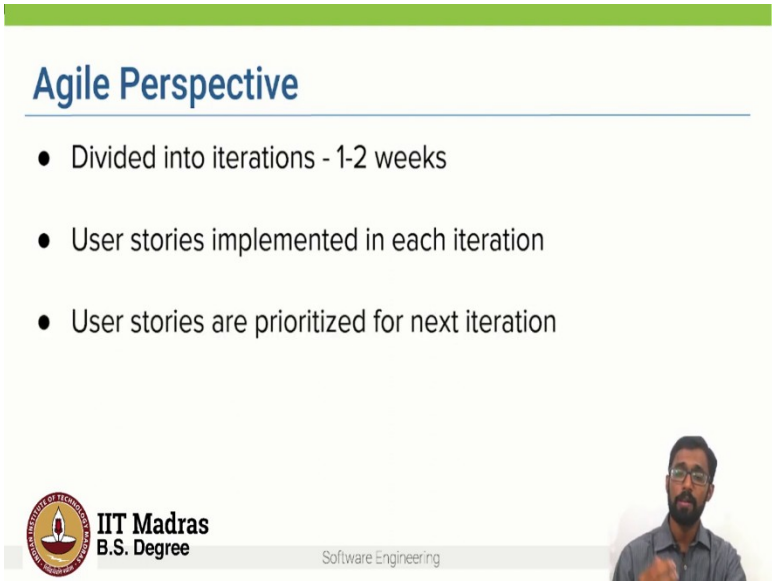
(Refer Slide Time: 00:20)




In the previous videos of this week, we looked at key activities in managing a software project. We looked at activities such as project planning, estimation, risk management. And these are common practices and tasks in the Plan and Document perspective. And if you recall, in the plan and document perspective, the goal has been to make software engineering, predictable in budget and schedule. So, there is a lot of documentation and planning at the start of the project.

However, if you look at it from the Agile perspective, Agile does not try to predict cost and the schedule at the start of the project itself. Instead, it relies on working with customers on frequent iterations and then agreeing on a time and how much effort and cost is required.

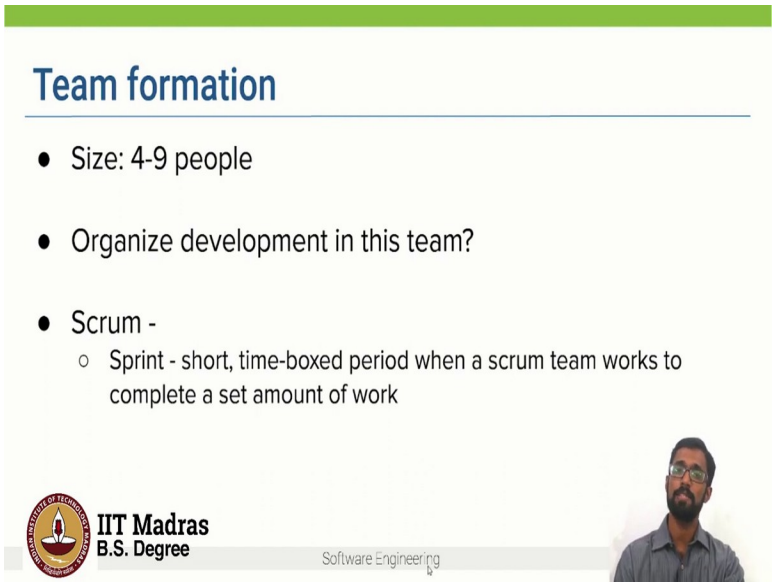
(Refer Slide Time: 01:25)




Now, let us just recall the Agile perspective and the philosophy which we discussed in the first week. So, the Agile lifecycle is divided into iterations of typically 1 to 2 weeks. And in each iteration, certain user stories are implemented. That is a working prototype is created from these user stories. And then users examine the work done in this iteration to see if it matches the requirements. And then user stories are prioritized for the next iteration.

New user stories are created as well. And then this process continues till the entire system is built. So, in this video, we will be looking at how work is actually done from the Agile perspective, how is planning done, how is she doing done? And what all happens for this work to get done?

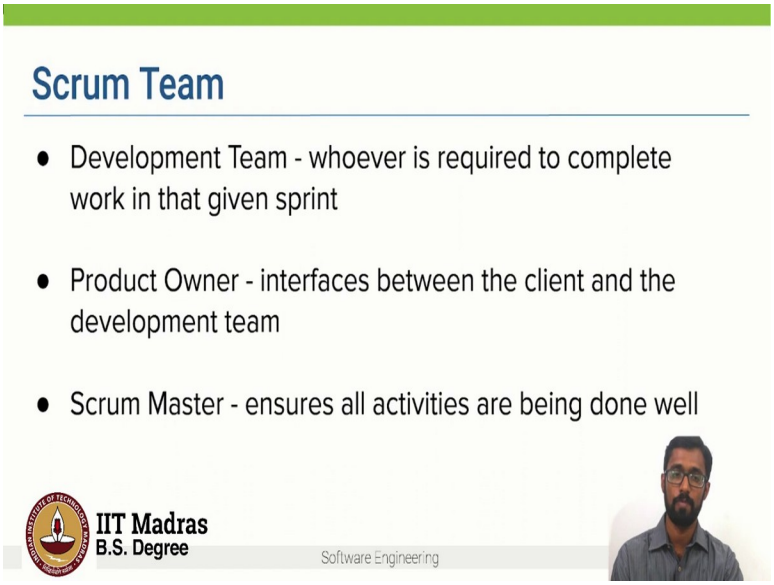
(Refer Slide Time: 02:32)



So, let us, look at how teams are formed. So, as we saw in the plan and document perspective, one of the first jobs of the project manager is to form and organize team. So, in the Agile perspective, the team size is roughly 4 to 9 people for a particular project. Of course, this varies from company to company. And now the question is, if you have the team now, how do you organize development in this team? And there are various strategies and frameworks. And we will be looking at one of them in this video, which is known as Scrum. And let us see what scrum looks like.

So, at the heart of Scrum, is a sprint, a sprint is short time box period when a scrum team works to complete a set amount of work, so there is a team, which works in what is known as different sprints, or iterations. And with scrum, this product is built in a series of iterations which are known as sprints. And in this sprints, the complex project is broken down into small pieces. Now, let us look at the main roles in a scrum team.

(Refer Slide Time: 04:07)



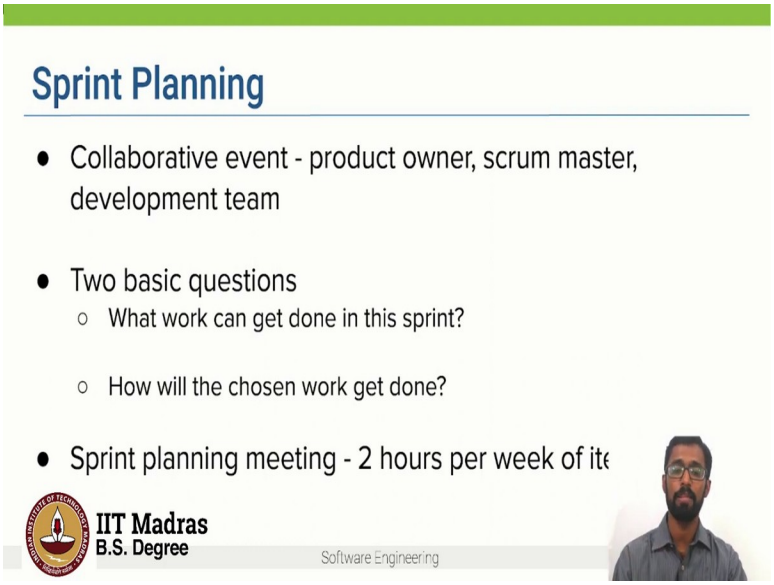

So, there are three main roles in a scrum team. One is the development team. So, the people who do the actual work of developing the software. And many of you might think of only coders or developers in the development team but this is not so. So, whoever is required to complete the work in that given sprint, they can form a part of the they form the part of the development team. So, it can be designers, developers, testers, who form a part of the development team.

The second role is that of a product owner. The product owner is a person who interfaces between the client and the development team. And the product owner ensures that the team is

delivering the most value to the client. And the product owner also takes into account other stakeholders in the organization. For example, the constraints of developer's, technology constraints etc. and communicates all of this to the client.

So, the most important responsibility of the product owner is to take all these inputs from the clients and the development team and prioritize the work to be done in a particular sprint. And the third role is known as a scrum master. The scrum master ensures that all the activities in the scrum are being done well. He or she helps the product owner define value and the development team to deliver the value to the product. The scrum master also serves the organization at large helping the organization understand what scrum is and create an environment that supports scrum. Now, let us look at the main activities in scrum.

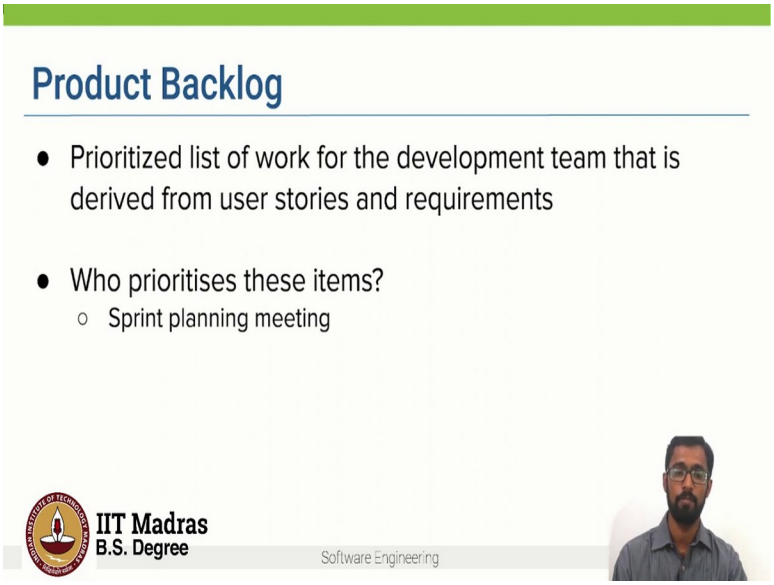
(Refer Slide Time: 06:26)




So, before you start a sprint, you conduct a meeting known as a sprint planning meeting. So, this is a collaborative event between the product owner, the scrum master and the development team. And the team asked two basic questions about what work can get done in this sprint and how will the chosen work get done?

So, the team chooses a subset of all the tasks which are required to be done and decides on which tasks have to be done in this sprint. And the sprint planning activity or meeting is time boxed and it is roughly two hours per week of the iteration. So, then, how do we know what all are the tasks which are required to be done in a particular sprint or what is the entire set of tasks for developing that product.

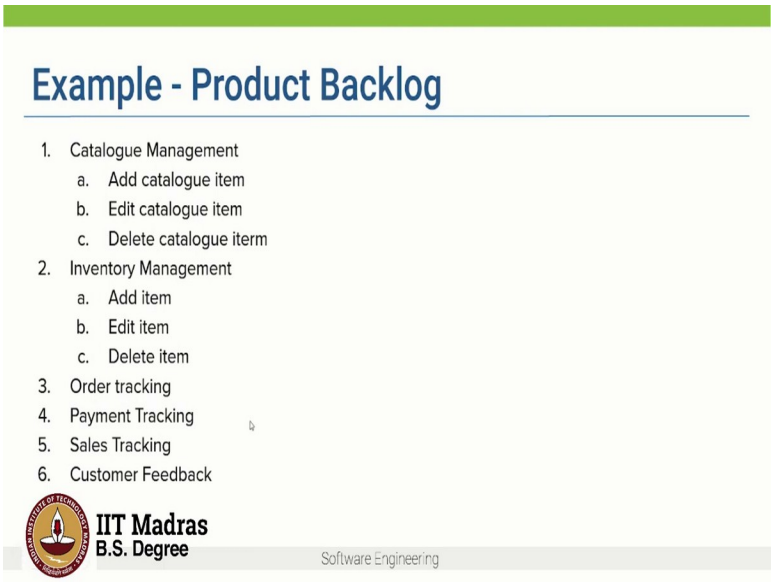
(Refer Slide Time: 07:40)




So, that is where the product backlog comes in. So, a product backlog is a prioritized list of work for the development team. And this is derived from the user stories and requirements. So, the most important items are shown at the top of the product backlog so that the team knows what has to be delivered first. So, now, who prioritizes these items in the product backlog? So, it is not the product owner.

But in the sprint planning meeting, all the team members collectively come up with the items which will be worked on in this sprint in a particular sprint. So, for example, the product owner may say, the client requires these user stories immediately. But the development team may state their constraints and then the entire team would come to a consensus.

(Refer Slide Time: 08:53)



Now, let us look at an example of a product backlog. Let us take the example of the Amazon Seller portal itself. And in previous weeks, we have come up with some user stories based on the requirements. And this involved in managing the catalog, managing the inventory, tracking the orders, payment, sales, etc. And now we have to decide for a particular sprint, what are the items we need to work on? So, let us say for the first sprint, the developers say okay, we can work on catalog management. Maybe we can work on 1a and 1b of adding and editing catalog items.

But then the product owner tells well actually the client wants to see catalog management as well as inventory management. So, then, the developers might say that it is not possible to complete both. In this sprint, maybe we can focus on adding a catalog item and maybe adding an item in the inventory. So, we can focus on 1 a and 2 a. So, the entire team comes to a consensus and decides to work on 1 a and 2 a in this sprint.

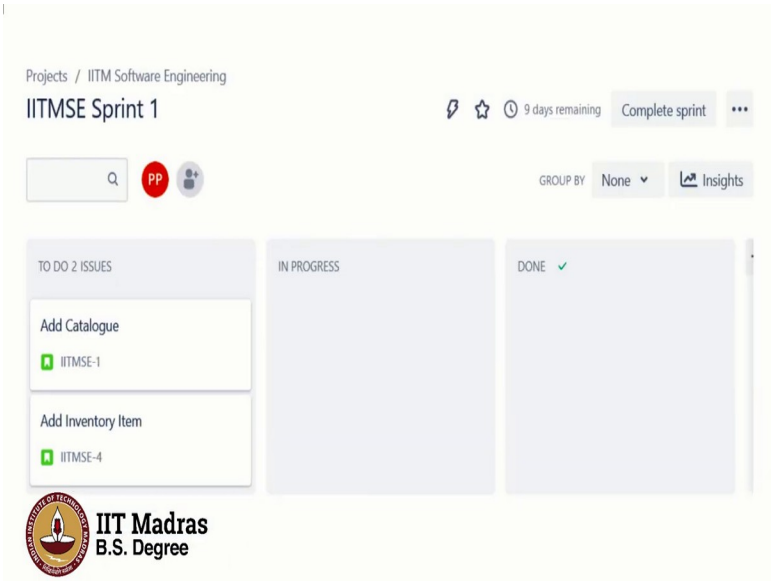
(Refer Slide Time: 10:26)


So, this image is taken from a popular software management software which is known as JIRA. So, in JIRA, I can create a backlog which has all these issues or these are known as the items in the backlog and this is similar to what we saw in the previous screen.

(Refer Slide Time: 10:54)

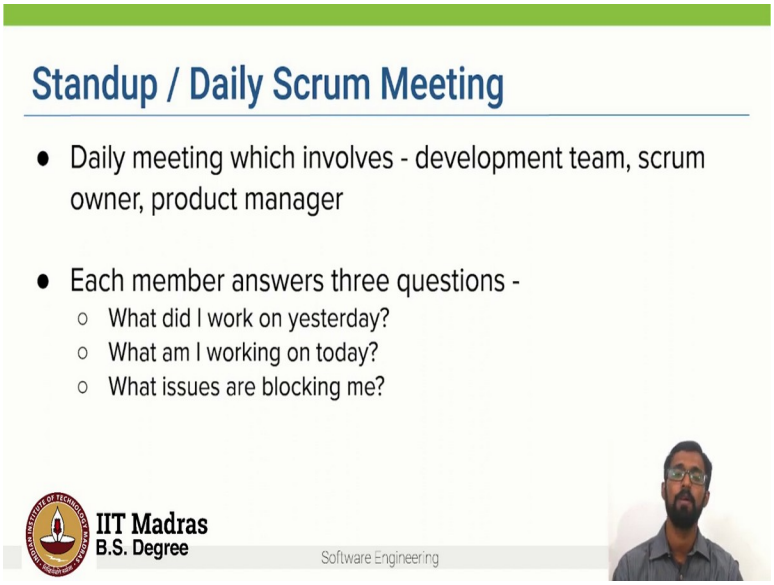

And then, for a particular sprint, I can create sprints and I can take items from the backlog and add it to the sprint. And then I can decide a timeline for the sprint. I can assign specific user stories to specific people. And all this can be done in most software project management tools.

(Refer Slide Time: 11:25)




And by the end of the sprint planning meeting, now the team has decided which are the issues which will be worked on in this particular sprint. And now the team is ready to start work on the sprint backlog taking items from the backlog to the in progress and the done tabs. So, here you can see that there are three boards. And as we saw in the previous screen, we have decided to work on two items. And these two items are present in the to do list for the sprint for sprint 1.

(Refer Slide Time: 12:08)

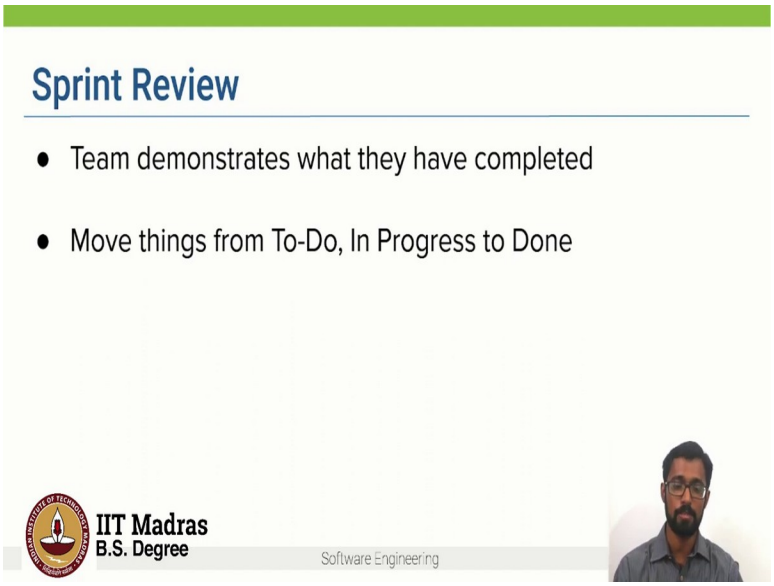



So, now that we have seen what happens in a sprint planning meeting, and we know what are the items to be worked on for a particular sprint. Now, how does the team make progress for in this particular sprint and this is done using a standup or a daily scrum meeting. So, this meeting is a daily meeting which involves the development team, the scrum owner, a scrum

master and the product manager. And basically each member of the team answers three key questions.

One is what did I work on yesterday? What am I working on today? And what issues am I facing? Or what issues are blocking my progress? And when each member answers these questions, it strengthens the team because everyone shares the progress of how they are contributing to the team. So, this daily scrum meetings continue till the end of the sprint, and hopefully by the end of the sprint, the team would have developed the prototype or the features for that particular sprint.

(Refer Slide Time: 13:38)

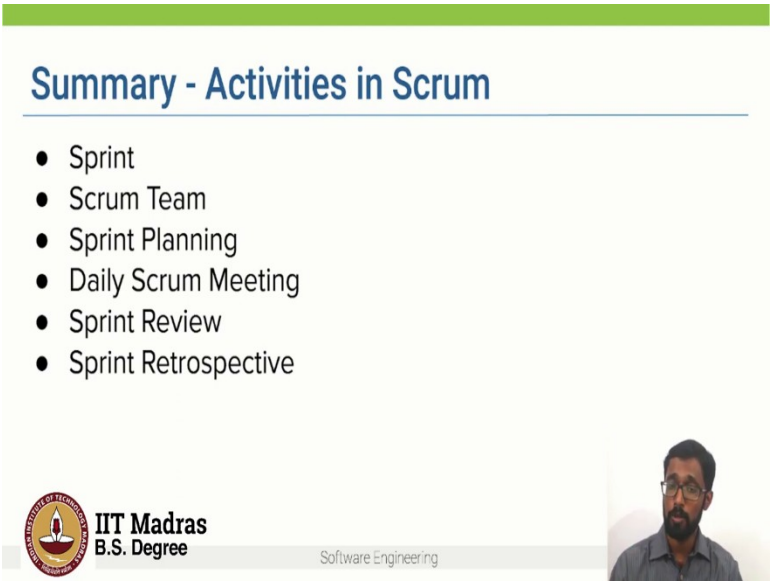



And this progress is reviewed in the sprint review meeting. So, after the sprint, the team demonstrates what they have completed during the sprint review meeting. So, what all things were there in the to-do or the in progress buckets, they would have ideally been moved to the done bucket. And the team demonstrates what they have completed in that sprint.

(Refer Slide Time: 14:11)


And finally, there is a sprint retrospective meeting. And then this serves several purposes. So, one is to evaluate the last sprint in terms of the team dynamics, how did the team work, the processes, the tools. So, it is basically an evaluation of how things went in the previous sprint. And also discuss and talk about which user stories or tasks went well or did not go well do well. So, what are the reasons behind that? And finally, creating and implementing a plan to improve how the team does work so that it will improve in the upcoming sprints.

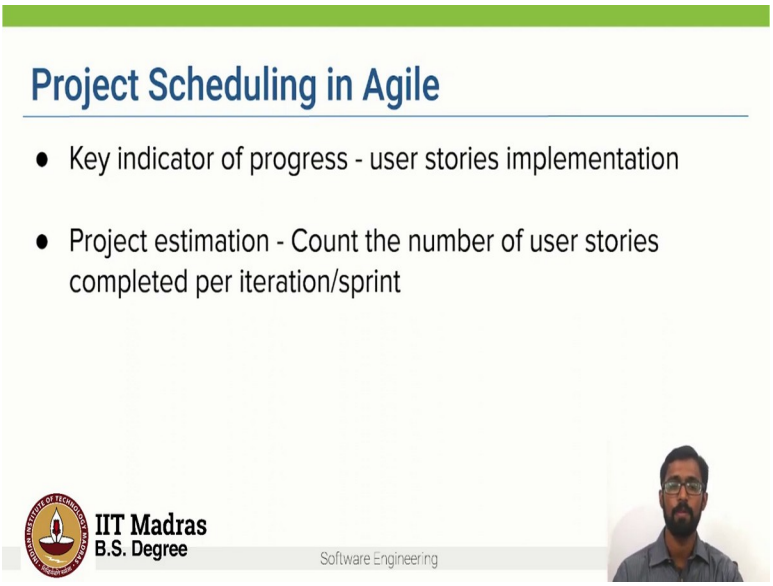
(Refer Slide Time: 15:06)




So, to summarize, we looked at a framework known as scrum in this video, and scrum has several activities, the key being the sprint, which is short, one to two-week duration in which certain items are decided on and the team works on completing those items. We looked at the

scrum team, we looked at several meetings or several activities in the scrum, which is the sprint planning, the daily scrum meetings, and the sprint review and the sprint retrospective meetings, which are conducted after the sprint ends.

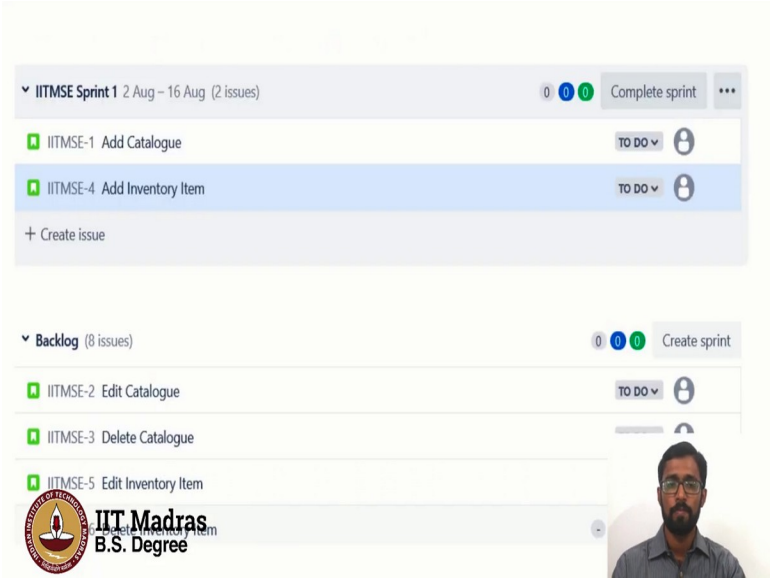
(Refer Slide Time: 15:54)




Now, let us look at how we can do project scheduling in Agile. So, if you recall, in the plan and document perspective, we looked at Project Scheduling techniques like breaking down the activity into tasks and creating representations like a Gantt chart. So, in the Agile methodology, we do it slightly differently. As we saw in this video, work is done in sprints or iterations of typically 1 to 2 weeks. And in each iteration, certain user stories are implemented.

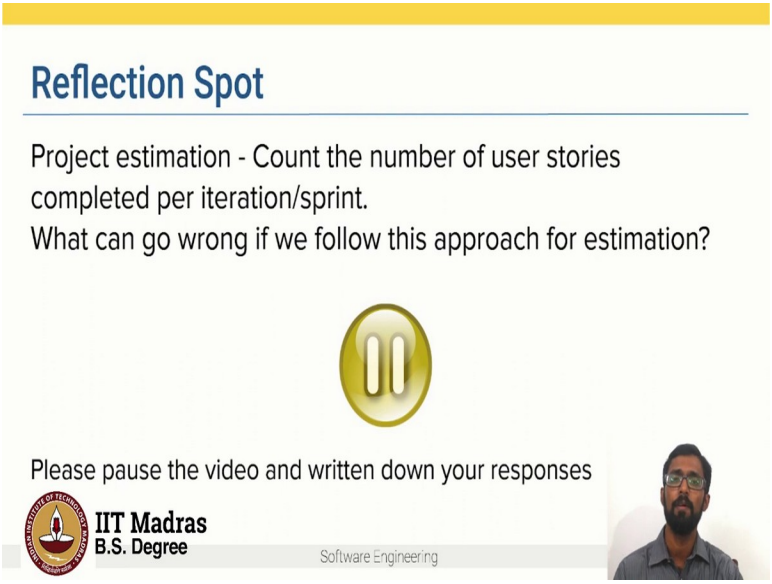
So, the key indicator of progress is how many user stories have been delivered to the client then each iteration. So, we can do project estimation by simply counting the number of user stories completed per iteration. And then calculate the average number of stories per week. And this will give us an estimate of how much time effort and cost is required to complete the entire project.

(Refer Slide Time: 17:06)




So, let us take the example of the seller portal. So, we previously identified that we require or we can complete two user stories in sprint 1. And we have a total of 8 plus 2, 10 user stories. So, we can complete the entire set of 10 user stories in 5 iterations, which is nothing but and let us say each iteration takes 2 weeks. So, in 10 weeks, we can complete the entire project. So, this helps us define the schedule for the project. So, now let us just reflect on this process, which we did.

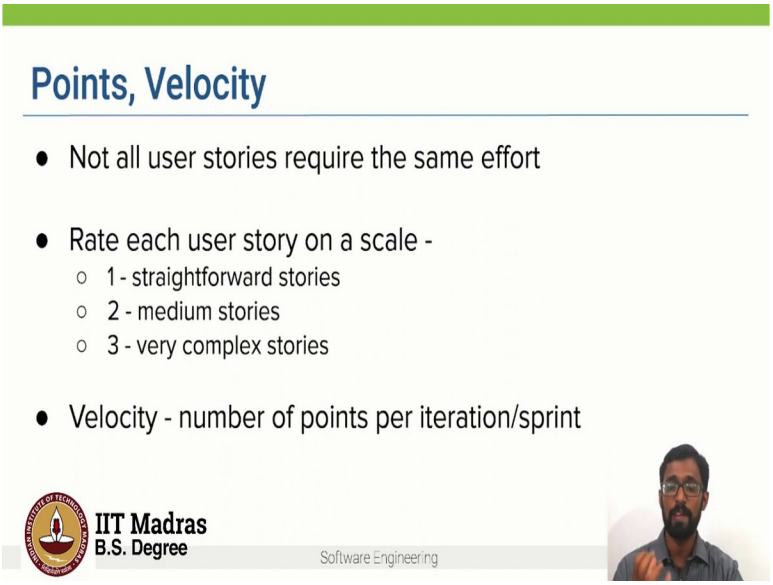
(Refer Slide Time: 17:53)




So, what we are saying is that, by counting the number of user stories completed per sprint, we can estimate the time required to complete the project. So, do you think this is a good

idea? So, what can go wrong if we follow this approach for estimation? You can pause this video and think about some issues related to this approach and then proceed.

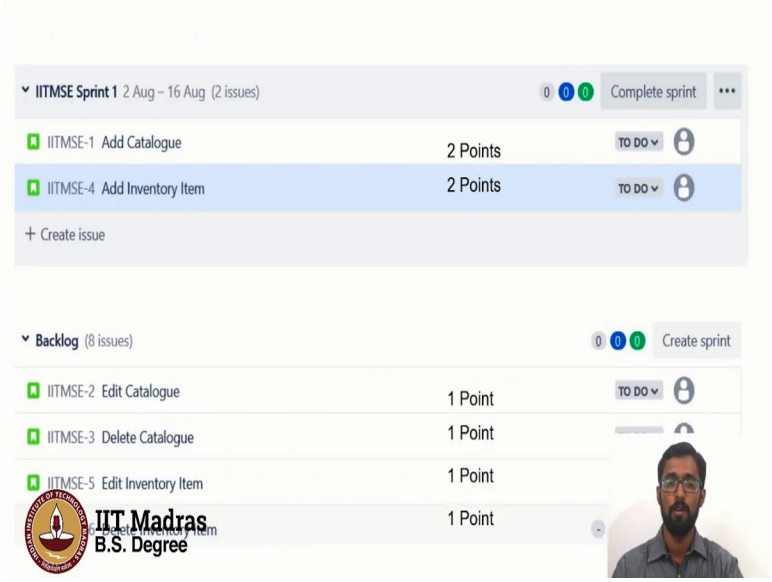
(Refer Slide Time: 18:29)




The key issue is that not all user stories require the same effort. Some are much harder to implement than others. And this can lead to miss predictions. So, one solution can be to rate each user story on a scale. So, we say the rating for a user story is one if it is a very straightforward and a simple user story. Two, if it is of medium difficulty or effort. And three, if it is a very complex user story.

So, now we have assign points to each user story. And now we can calculate what is known as a velocity which is nothing but the number of points per iteration or sprint. And this term velocity in a way it measures the work rate of the team. In one iteration, how many points is the team able to complete? And this helps stakeholders also to get an idea of how many iterations it will take a team to add the desired set of features.

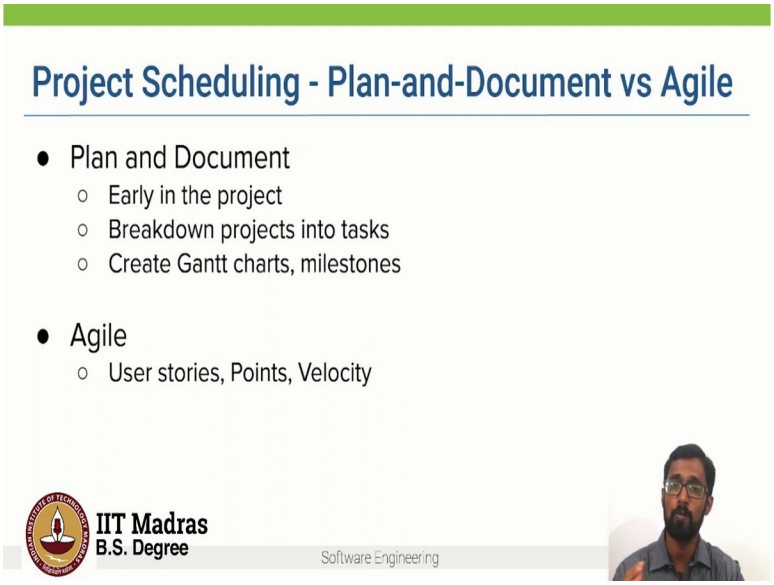
(Refer Slide Time: 20:00)




For example, in the seller portal, the team assigns points to each user story. So, here we can see that add catalog is assigned 2 points, add inventory item is also assigned 2 points. And some other user stories are assigned 1 point. And now the team decides to work on these two user stories in sprint 1. And let us say after the sprint, they have successfully completed these 2 items. So, we see that the velocity of this team is 4, 2 plus 2. So, in a sprint, this team is able to complete user stories, totaling 4 points,

So, now in the next iteration, instead of just working on two user stories, the team can very well work on all 4, because it requires 4 points to complete. And we know that the team can complete 4 points in a single iteration. So, in this way, project scheduling for an agile development process can be done using points and velocity.

(Refer Slide Time: 21:26)




So, now we just compare the project scheduling in the plan and document and the agile perspectives. So, in the plan and document perspective, it is done early in the project at the start, where we break down the project into tasks. And then we create Gantt charts and milestones, which define the schedule of the project. Whereas what we saw in the agile perspective is that we create user stories. And we assign points to this user stories. And we calculate velocity to and this where this term velocity helps us scheduled the iterations and sprints, and how much time the project will take to get complete.