

UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Definition of Done

- The Agile definition of done is a collection of criteria that must be completed for a project to be considered “done.” It is essentially a checklist used by Scrum teams to create a shared understanding of what is required to make a product releasable.
- To fully understand the definition of done in the context of Scrum, we must first outline two of the key elements in Agile project management:
- **Product backlog item:** This is a specific improvement that is made to a product. Items can include bug fixes, user stories, and specifications.
- **Product increment:** This is what is produced at the end of a short development period or sprint. It combines all the product backlog items completed during this sprint.
- In specific terms, the Scrum definition of done is a list of conditions that must be met to successfully mark a product increment as complete.



Definition of Done

- The Agile definition of done is a collection of criteria that must be completed for a project to be considered “done.” It is essentially a checklist used by Scrum teams to create a shared understanding of what is required to make a product releasable.
- To fully understand the definition of done in the context of Scrum, we must first outline two of the key elements in Agile project management:
- **Product backlog item:** This is a specific improvement that is made to a product. Items can include bug fixes, user stories, and specifications.
- **Product increment:** This is what is produced at the end of a short development period or sprint. It combines all the product backlog items completed during this sprint.
- In specific terms, the Scrum definition of done is a list of conditions that must be met to successfully mark a product increment as complete.



Definition of Done

- Implementing the Agile definition of done offers some key advantages, which coincide with the three pillars of Scrum:
- **Transparency:** Everyone on the Scrum team has a clear understanding of what constitutes “done.” This eliminates confusion and potential clashes of opinion because there is a pre-defined set of rules for what the product should look like when complete.
- **Inspection:** This stage is carried out by every team member throughout the sprint, so issues can be spotted early and addressed quickly. The team can also share the product with customers for testing purposes to gather feedback and ensure their needs are being met.
- **Adaptation:** After the inspection stage, improvements can be made to reach the definition of done in Scrum. These adaptations will ensure higher quality in the finished product, potentially increasing ROI and boosting customer satisfaction.



Agile Design

- Design plays a vital role in any software development process. The agile team also focuses on "what to do about design" because of the following four factors:
- Many crucial factors focus on loyal designs during the planning process. Design forces towards waterfall culture throughout product implementation.
- Designers also interact with a cross team for a limited time.
- Designers don't always have an easy way to report feedback to the engineering team.
- The presentation and logic layers are not still transparent. They are not separated clearly in the code base, making style changes difficult.
- Agile design is a way to understand the idea or methodology of agile, how it runs, how the work is carried out in this, and all other necessary things.



Incorporation of Design Process

- The agile design process uses the iterative as well as incremental approach; in this particular process, the system divides the work into small parts and puts the focus on individual parts holistically. This approach includes:
- **Feedback:** In this process, one can easily interact with the customer to know his requirements and also get the feedback of the client or customer regarding the product and make necessary changes required according to the feedback; it is a very helpful function in the agile design.
- **Changeable:** During the agile design, it is easily doable if any changes are required in the design. There is the proper function for the changes, and software is very helpful for the alteration in the design.
- **Development:** The development in agile is very fast; teams need not waste much time on the allied processes; its easier functions and smooth functioning make the process speedy.



The Methodology of Agile Design

- **Scrum:** It is the widely used methodology in the market when anyone thinks about working in agile; this is the first thing that came to his/her mind. In this methodology, both the planning and implementation of the project happens at the same time. Therefore, it is the easier one to work in agile.
- **Crystal:** This particular methodology focuses on the key areas of the project and concentrates on the project's major priorities. Also, the satisfaction of the customer is the prime agenda of this methodology.
- **Agile UX:** The UX version or this type of methodology focuses on the outcome of the project rather than the negative views of whether the product is successful and all that.
- **Extreme:** The extreme methodology is used mainly for software development and focuses on the end product, and ensure that it should be as far as the customer needs.



Process of Agile Design

- **1. Design:** The design of agile will be made by using any of the methodologies used above, but the best design is that which is customer or client-centric and gives fruitful results. Also, the team that takes charge of making a better agile design pays proper vigil on the project so that no time and resources will be wasted. The design is a hectic task at the inception of the project, which needs proper attention.
- **2. Analyze:** When the design is completed and work is on track, a proper analysis is required from time to time so that the scope of the fault should be eliminated and the quality of the product is maintained. The analyses of the design are also helpful for the team to complete work on time.



Process of Agile Design

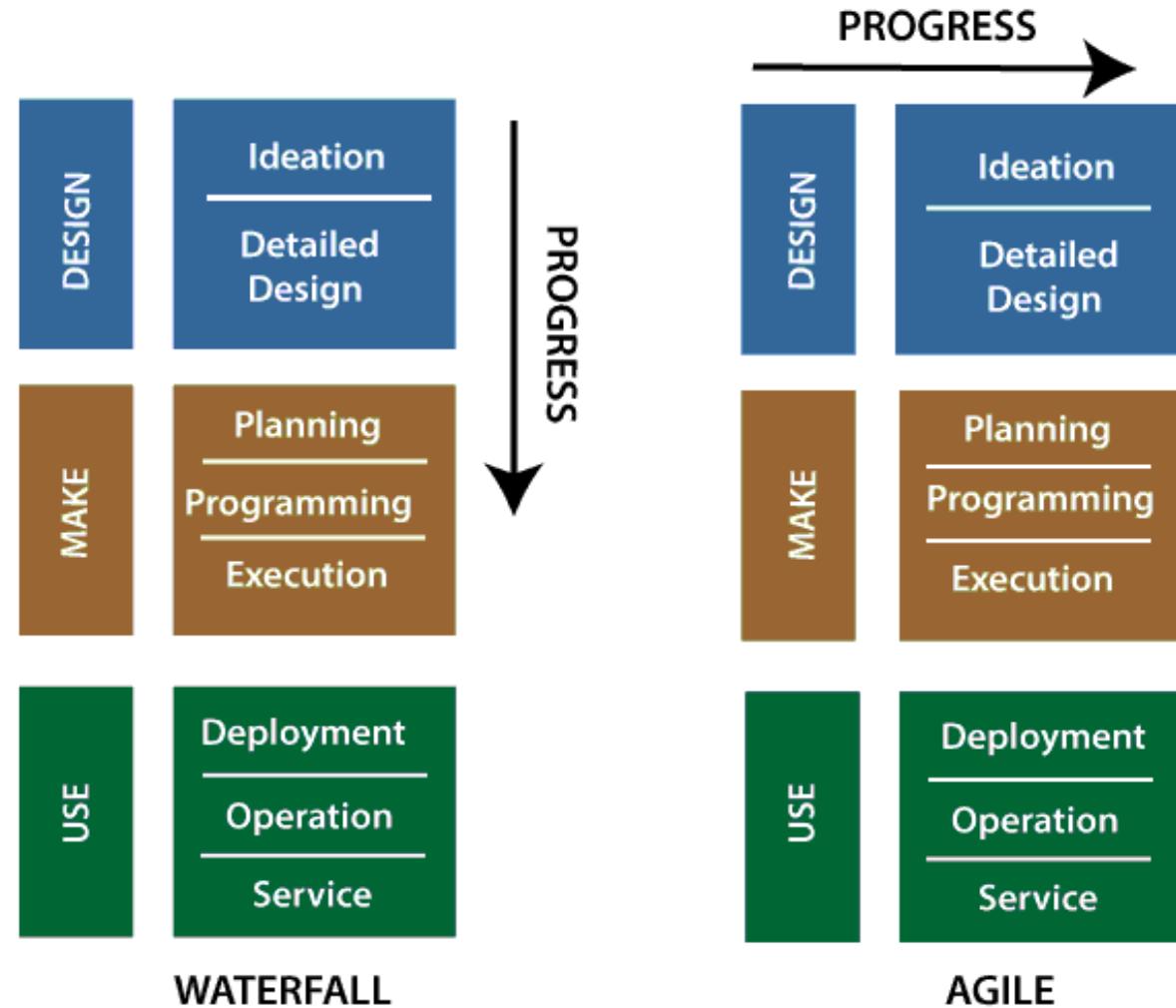
- **3. Develop:** Here, develop means to develop the required project management software, which is the prime moving force of the project after its completion. Management is very necessary after the project execution, so agile management is much required.
- **4. Implement:** The last stage of the process is to implement the agile project and its execution. The feedback of the design is most important so that the team can make changes if required. Also, the satisfaction of the client is much-required things after the implementation.

The product design process and customer interview

- The agile is divided into several methodologies and processes. These methodologies and processes keep the iterative and free-flowing nature of the technique at their core.
- The agile design and development methodology used especially in engineering development, and this process called Scrum.



The product design process and customer interview



The product design process and customer interview

- Customer interviews can be an informative part of the project design phase. We will have several of those "light bulb" movements during interviews. It encourages the people who are interviewing with other members of the team (engineering, marketing, design, etc.)
- There are several resources that are available on which we conduct an interview-the logistics, methods, and techniques.



The customer interview pyramid

- Atlassian is a simple framework that helps in building the customer interview pyramid. This pyramid looks like as
-



The customer interview pyramid

- **Communication Observation:** At the bottom of the pyramid, we will get the very minimum. We should all come back from an interview and be able to list observations as we don't need any experience to regulate what you've seen.
- **Interpret problems:** Above the Communication Observation, it is an interpret problem. It is explaining the user's behavior and grouping them with an over-arching problem statement.
- **Connecting opportunities:** This is the peak of the pyramid where the most value comes in combining the problem with potential opportunities or related patterns. This helps influence a roadmap and make decisions about what to tackle next.
-





THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Daily Stand-up

- A daily stand-up is a daily status meeting among all team members and it is held roughly for 15 minutes.
- Every member has to answer three important questions –
 - What I did yesterday?
 - What I'll do today?
 - Any impediment I am facing.../ I am blocked due to...
- Daily stand-up is for status update, not for any discussion. For discussion, team members should schedule another meeting at a different time.
- Participants usually stand instead of sitting so that the meeting gets over quickly.



Daily Stand-up

DAILY STAND-UP MEETING



Time Box (15 min)



Same place



Same time



Facilitated by Scrum Master



Full team presence



Focus upon 3 questions

3 Main Questions

1. What I did yesterday?
2. What I'll do today?
3. What's in my way?



Why Stand-up is Important?

- The benefits of having a daily stand-up in agile are as follows –
- The team can evaluate the progress on a daily basis and see if they can deliver as per the iteration plan.
- Each team member informs all about his/ her commitments for the day.
- It provides visibility to the team on any delay or obstacles.



Who Attends a Stand-up?

- The scrum master, the product owner, and the delivery team should attend the stand-up on a daily basis.
- Stakeholders and Customers are encouraged to attend the meeting and they can act as an observer, but they are not supposed to participate in stand-ups.
- It is the scrum master's responsibility to take note of each team member's queries and the problems they are facing.



Geographically Dispersed Teams

- Stand-ups can be done in multiple ways, in case the agile team members are operating from different time zones –
- Select a member on a rotational basis, who can attend the stand-up meeting of teams located in different time zones.
- Have a separate stand-up per team, update the status of the stand-up in a tool such as Rally, SharePoint, Wikis, etc.
- Have a wide variety of communication tools ready like conference call, video conferencing, instant messengers, or any other third-party knowledge sharing tools.



Geographically Dispersed Teams

- Stand-ups can be done in multiple ways, in case the agile team members are operating from different time zones –
- Select a member on a rotational basis, who can attend the stand-up meeting of teams located in different time zones.
- Have a separate stand-up per team, update the status of the stand-up in a tool such as Rally, SharePoint, Wikis, etc.
- Have a wide variety of communication tools ready like conference call, video conferencing, instant messengers, or any other third-party knowledge sharing tools.



Definition of Done

- **User Story**
- A user story is a requirement which is formulated in a few sentences in everyday language of an user and it should be completed within an iteration. A user story is done when
 - All the related code have been checked-in.
 - All the unit test cases have been passed.
 - All the acceptance test cases have been passed.
 - Help text is written.
 - Product Owner has accepted the story.



Definition of Done

- **Iteration**
- An iteration is a time boxed collection of user stories / defects to be worked upon and accepted within the release of a product. Iterations are defined during iteration planning meeting and completed with an iteration demo and review meeting. An iteration is also termed as a **sprint**. An iteration is done when
- Product backup is complete.
- Performance has been tested.
- User stories have been accepted or moved to the next iteration.
- Defects have been fixed or postponed to the next iteration.



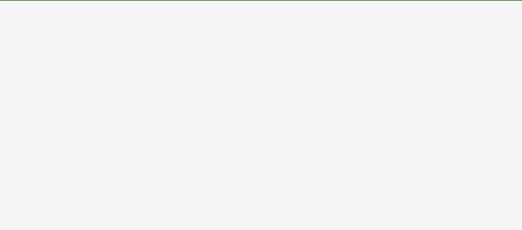
Definition of Done

- **Release**
- A release is a major milestone that represents an internal or external delivery of working, tested version of the product/system. A release is done when
- System is stress tested.
- Performance is tuned.
- Security validations are carried out.
- Disaster recovery plan is tested.





THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Scrum Workflow

- Scrum is a framework that addresses complex problems, while creatively and productively delivering products of the finest possible value. It is developed for teams who divide their work into goals, that can be completed within timeboxed iterations called sprints, which lasts 2- 4 weeks.
- The scrum team tracks the progress of the project in 15-minute timeboxed meetings called daily Scrum. After sprint completion, the team holds a review meeting to demonstrate the work done and a retrospective to continually improve.
- Let's get to the next part of this “Scrum Workflow” article and learn about the important roles involved in the Scrum framework.



Scrum Roles

- **Product owner** - The product owner in the Scrum team is primarily responsible for working with the user group to determine the features to be included in the product release. They are the champions who manage the team and guide the project's progress toward success.
- **Scrum Master** - The Scrum Master in the team is responsible for promoting and supporting Scrum. They guide the team, product owner, and business on Scrum and look for ways to fine-tune their practice. An effective scrum master profoundly knows how the work is being performed by the team and helps the team in optimizing their transparency and delivery flow.
-



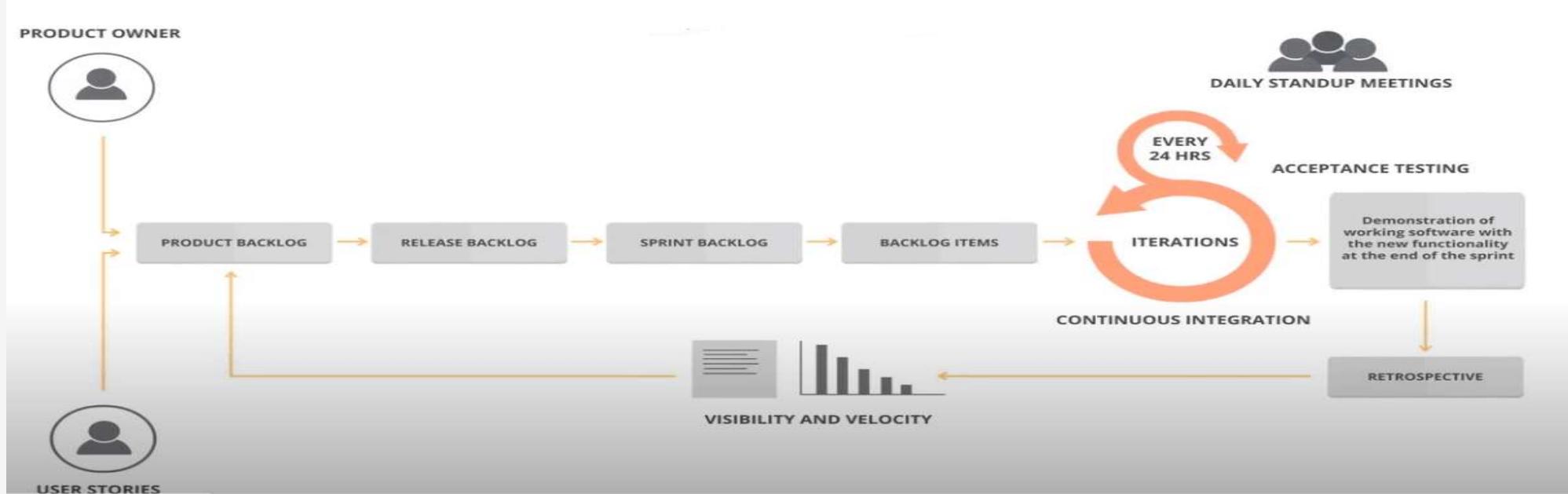
Scrum Roles

- **Development Team** - The development teams are cross-functional and have the skills to deliver product increments. This team includes developers, testers, designers, etc., to have less dependency on the third party. All the members of the scrum team are self-organizing and ensure successful sprint completion by turning Product Backlog into increments of potentially releasable functionality.



Scrum Workflow Steps

- The below diagram depicts the step by step process involved in scrum workflow:



Scrum Workflow Steps

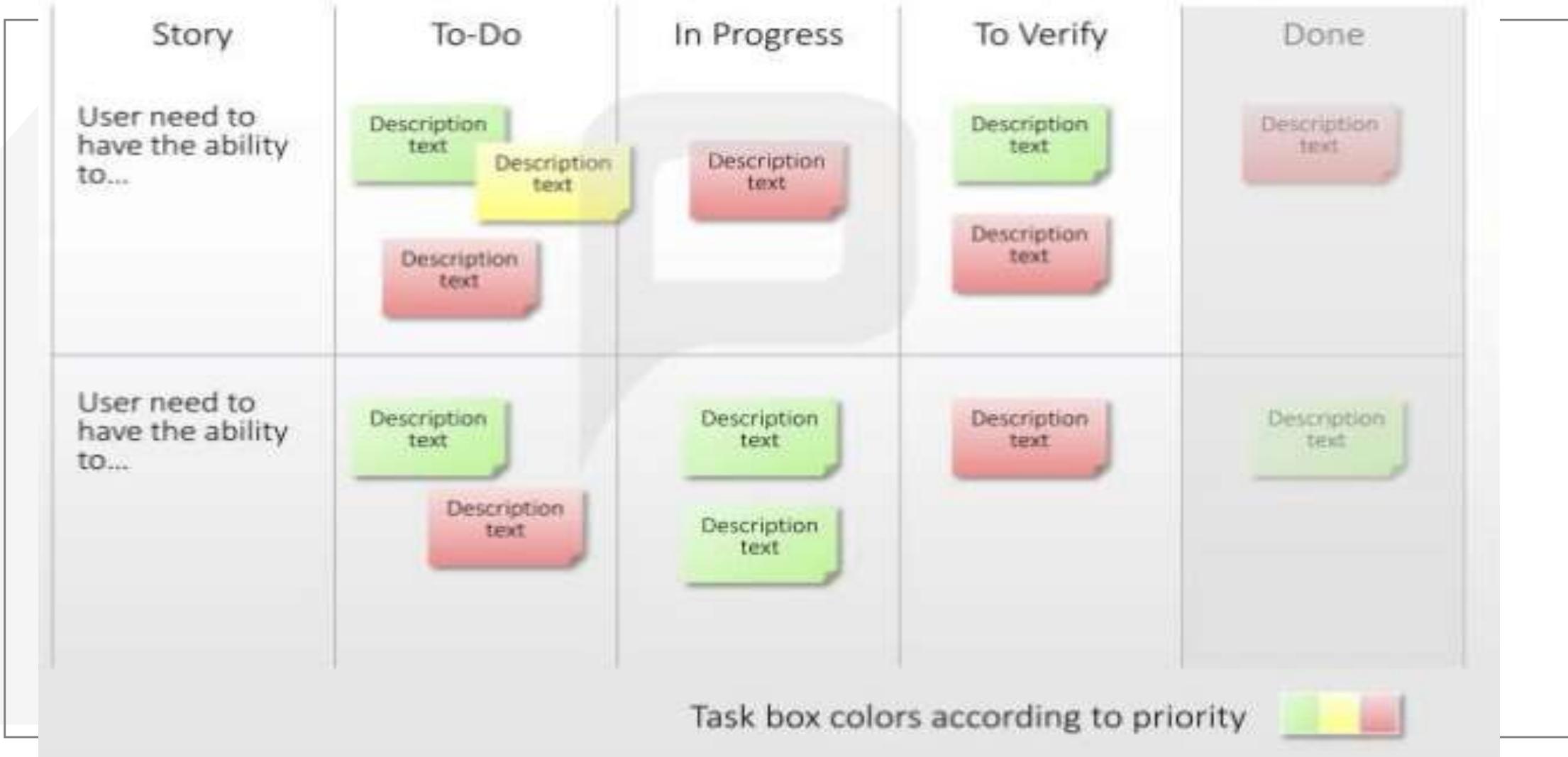
- **1. Product backlog creation**
- The first phase of the Scrum workflow process begins with the visioning phase, where the Stakeholders meet to decide the list of features that should be implemented and develop a product roadmap.
- **2. Release backlog**
- Based on the product roadmap developed, in collaboration with the product owner, the team decides how to group user stories into releases. The objective of release is to deliver a subset of product backlog known as release backlog



Scrum Workflow Steps

- **3. Sprint Backlog creation**
- A Sprint is a predefined timeframe within which the team performs a set of tasks from the Backlog. The duration of each Sprint lasts 2-4 weeks. Each Sprint takes a manageable chunk of release backlog and gets it a ship steady state. A set of product backlog items that must be delivered within a single sprint iteration is called Sprint backlog.
- **4. Working on sprint and scrum meetings**
- After the user stories for the current phase are selected, the development process begins. For tracking the current working process, a task board is commonly used, which represents particular user stories with a description of tasks needed for implementation.







THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

1. Iterative Development

- Through agile iterative development, bigger projects are broken down into smaller chunks and continuous tests are done in repetitive cycles. Through this practice, agile teams get a perspective on new features that need to be added to the final product or service and contributes towards more flexible product development.



Agile Practices

2. Daily Meetings

Regular meetings are key to agile implementation. These meetings should be short and concise, with each member of the team explicitly stating the progress of tasks and what needs to be done. This practice is a great way to monitor the performance of the team and check if there are any obstacles in the way of product development.



Agile Practices

3. Using Professional Tools

- Using project management tools for the implementation of agile methodology helps the team to better structure their workflows and improve team collaboration. For proper documentation and meetings management, professional project management software can greatly reduce the effort it takes to manage your tasks otherwise.



Agile Best Practices: Scrum Project Management

1. Creating Product Backlog and Product Vision Together

A product backlog is an ordered list of items that are required to be added to product development. A good practice for scrum implementation is to create the product backlog and product vision together so that both the development team and stakeholders are on the same page. This ensures mutual understanding and helps in aligning the vision in a better way.



Agile Best Practices: Scrum Project Management

2. Use Burndown Charts for Sprints

- A daily burndown chart is a great way of monitoring the progress of Sprints. Burndown charts graphically show the work that has been done and the total work remaining against time. It's a useful tool to inform the team about project scope and make them aware of scope creep that might occur. These charts also help in identifying the risks associated with undelivered work.



Agile Best Practices: Scrum Project Management

3. Setting communication guidelines for teams

- Uninterrupted communication is key for the Scrum framework and can become a bottleneck if not tackled efficiently. An effective way to ensure seamless communication is to formulate a communication strategy with all the essential guidelines for teams. This particular practice can really come in handy for remote teams as it will make team goals transparent.



Agile Best Practices: Scrum Project Management

4. Practicing Stand-Ups

- Also known as the ‘Daily Scrum’, stand-ups are short meetings held with the team members on a daily basis. These meetings are typically for a maximum of 15 minutes to keep their duration short. Practicing Stand-ups for product or project development are a great way to monitor the progress of work and helps in keeping everyone in the loop with the project updates. These meetings also assist the team in tracking the dos and don’ts of product development.



Agile Best Practices: Kanban Project Management

1. Visualizing Workflows

- Visualizing workflows in the form of boards or cards showing the progress status of each task is an easy way to keep track of tasks and point out hurdles in the product development cycle. These boards generally come with the option of dragging and dropping the tasks from one pane to another to show the progress.



Agile Best Practices: Kanban Project Management

2. Limiting Work in Progress

- Fixed constraints for work in process limits the total amount of cards in the active pane, consequently helping the team in understanding the work that needs to be done within a stipulated timeframe. By limiting the unfinished work, a constant need for re-prioritizing tasks is eliminated and bottlenecks are identified more effectively.

Agile Best Practices: Kanban Project Management

3. Continuous Feedback

- Continuous feedback from team members is critical in order to understand how the team is going along with the process. These feedbacks also help in identifying any hurdles that might be occurring in the product development cycle and reflect on what needs improvement.



Agile Best Practices: Kanban Project Management

4. Focusing on Flow

- Monitoring the flow of work items assists the team in keeping an eye on overall work progress, giving them an idea of how quickly they need to move along with the process. This flow facilitates the team in understanding the speed and smoothness of delivery.



Lean Development Model

1. Identifying Value

- Carefully break down complex projects into smaller tasks and sub-tasks to identify the value associated with each one of them. This practice will lead to a better understanding of workflows and will help in identifying the unnecessary tasks that need to be eliminated, thus adding more value to the workstream.



Lean Development Model

2. Reducing Waste

- From the project management perspective, reducing waste addresses the elimination of any tasks, meetings, or documentation that are of no value to the overall product development. This elimination gives a clear direction to the team members and contributes towards the actual value addition process.



Lean Development Model

3. Continuous Improvement

- To successfully implement lean project management, there would be a constant need for improvements throughout the project development. One practice to achieve improvements is to clearly communicate the requirements and guidelines to team members for achieving more with minimal waste.



Extreme Programming (XP)

1. Planning Game

- All team members of a team should meet and participate in the planning process. There should be no ambiguity between the team(s) working on a particular project. This can take a form of meetings that occur after defined intervals to take updates and monitor progress accordingly.



Extreme Programming (XP)

2. Test-driven Development

- Before the final code, continuous tests are run to check the functionality of individual pieces of code. This practice helps programmers to go through situations where the code might fail. It also helps in lowering the defects and saves time to develop the software.



Extreme Programming (XP)

3. Small Releases

- Working on a similar principle of iterations, this concept focuses on small releases throughout the lifecycle of product development. This particular practice helps the entire team in understanding how the product is coming along, and identify any glitches that might occur during the product development cycle.

Extreme Programming (XP)

4. Simple Design

- The simple design of software requires less time to write and takes minimum effort to fix problems. This practice also helps in cutting down the overall costs of developing a product and paves way for team members to always find an easier way to get things done.



Process flow of Scrum Methodologies

PRODUCT OWNER



PRODUCT BACKLOG

RELEASE BACKLOG

SPRINT BACKLOG

BACKLOG ITEMS

EVERY
24 HRS

ITERATIONS

DAILY STANDUP MEETINGS

ACCEPTANCE TESTING
Demonstration of working software with the new functionality at the end of the sprint

CONTINUOUS INTEGRATION

RETROSPECTIVE



VISIBILITY AND VELOCITY



USER STORIES



THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Agile Kanban

- Agile Kanban is Agile Software Development with Kanban approach. In Agile Kanban, the Kanban board is used to visualize the workflow. The Kanban board is normally put up on a wall in the project room. The status and progress of the story development tasks is tracked visually on the Kanban board with flowing Kanban cards.

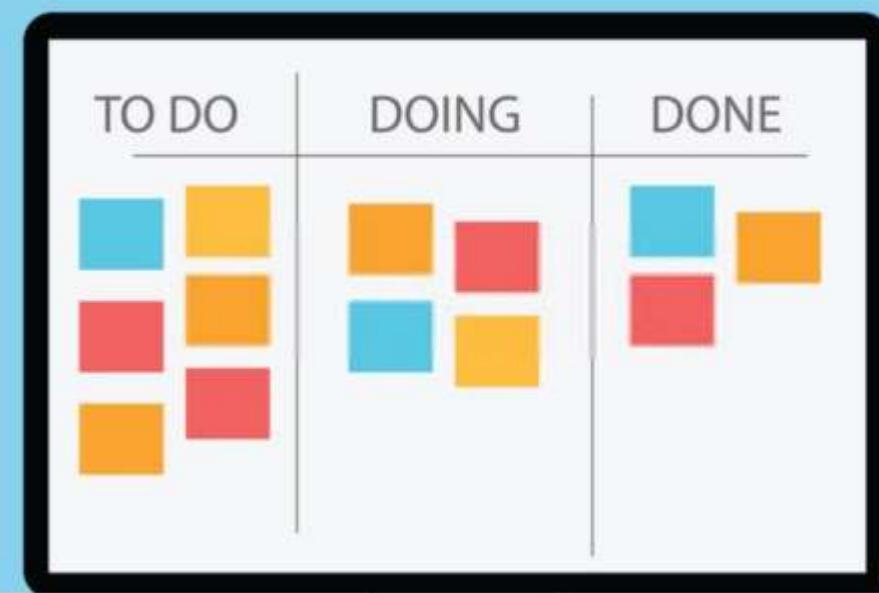


Kanban Board

- Kanban board is used to depict the flow of tasks across the value stream. The Kanban board –
- Provides easy access to everyone involved in the project.
- Facilitates communication as and when necessary.
- Progress of the tasks are visually displayed.
- Bottlenecks are visible as soon as they occur.



Kanban Board



Advantages of Kanban board

- The major advantages of using a Kanban board are –
- **Empowerment of Team** – This means –
 - Team is allowed to take decisions as and when required.
 - Team collaboratively resolves the bottlenecks.
 - Team has access to the relevant information.
 - Team continually communicates with customer.



Advantages of Kanban board

- **Continuous Delivery** – This means –
 - Focus on work completion.
 - Limited requirements at any point of time.
 - Focus on delivering value to the customer.
 - Emphasis on whole project.



WIP Limit

- The label in the Doing column also contains a number, which represents the maximum number of tasks that can be in that column at any point of time. i.e., the number associated with the **Doing** column is the WIP (Work-In-Progress) Limit.



Pull Approach and Self-directing

- Pull approach is used as and when a task is completed in the Doing column. Another card is pulled from the To Do column.
- In Agile Development, the team is responsible for planning, tracking, reporting and communicating in the project. Team is allowed to make decisions and is accountable for the completion of the development and product quality. This is aligned to the characteristic of empowerment of the team in Kanban.



Continuous Flow

- In Agile development, there is no gate approach and the work flows across the different functions without wait-time. This contributes in minimizing the cycle time characteristic of Kanban.

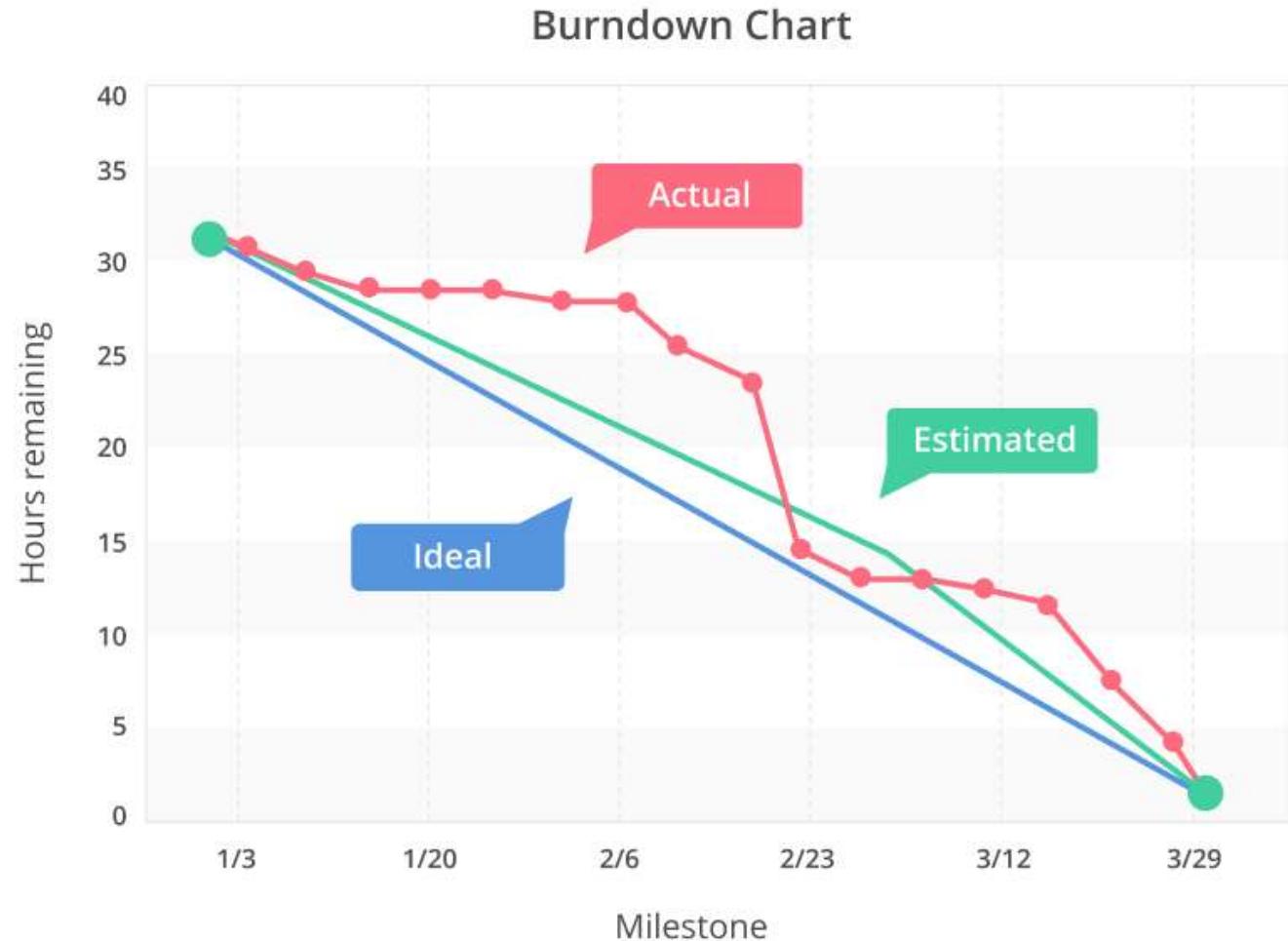


Visual Metrics

- In Agile Kanban, the metrics are tracked visually using –
- Kanban Board
- Burndown Chart
- **Uses of Kanban board**
- Kanban Board is used to –
- Measure the cycle times, that can be used to optimize average cycle time.
- Track WIP limit to eliminate waste.
- Track resource utilization to eliminate waste.



Visual Metrics



Visual Metrics

- **Uses of Burndown chart**
- Burndown chart is used to capture –
- The current status of the tasks and stories.
- The rate of progress of completing the remaining tasks.



Agile Vs Scrum

Agile	Scrum
Agile is a development methodology based on iterative and incremental approach.	Scrum is one of the implementations of agile methodology. In which incremental builds are delivered to the customer in every two to three weeks' time.
Agile software development has been widely seen as highly suited to environments which have small but expert project development team	Scrum is ideally used in the project where the requirement is rapidly changing.

Agile Vs Scrum

In the Agile process, the leadership plays a vital role.

Compared to Scrum it is a more rigid method. So there is not much room for frequent changes.

Agile involves collaborations and face-to-face interactions between the members of various cross-functional teams.

Scrum fosters a self-organizing, cross-functional team.

The biggest advantage of Scrum is its flexibility as it quickly reacts to changes.

In Scrum, collaboration is achieved in daily stand up meeting with a fixed role assigned to scrum master, product owner, and team members.

Agile Vs Scrum

Agile can require lots of up-front development process and organizational change.

Not too many changes needed while implementing scrum process.

The agile method needs frequent delivery to the end user for their feedback.

In the scrum, after each sprint, a build is delivered to the client for their feedback.

In this method, each step of development like requirements, analysis, design, are continually monitored during the lifecycle.

A demonstration of the functionality is provided at the end of every sprint. So that regular feedback can be taken before next sprint.

Agile Vs Scrum

Project head takes cares of all the tasks in the agile method.

There is no team leader, so the entire team addresses the issues or problems.

The Agile method encourages feedback during the process from the end user. In this way, the end product will be more useful.

Daily sprint meeting is conducted to review and feedback to decide future progress of the project.

Deliver and update the software on a regular basis.

When the team is done with the current sprint activities, the next sprint can be planned.

Agile Vs Scrum

Design and execution should be kept simple.

Design and execution can be innovative and experimental.

In the Agile method, the priority is always to satisfy the customer by providing continuous delivery of valuable software.

Empirical Process Control is a core philosophy of Scrum based process.

Working software is the most elementary measure of progress.

Working software is not an elementary measure.

Product backlog

- A product backlog is a list of items to be done. Items are ranked with feature descriptions. In an ideal scenario, items should be broken down into user stories.
- Why Product Backlog is Important?
- It is prepared so that estimates can be given to each and every feature.
- It helps in planning the roadmap for the product.
- It helps in re-ranking the features so that more value can be added to the product.
- It helps in determining what to prioritize first. Team ranks the item and then builds value.



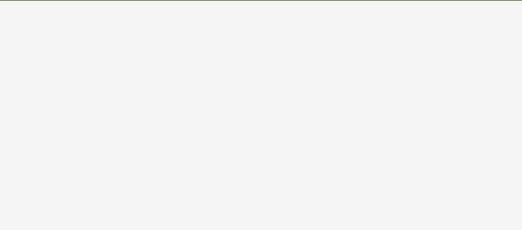
Characteristics of Product Backlog

- Each product should have one product backlog which can have a set of large to very large features.
- Multiple teams can work on a single product backlog.
- Ranking of features is done based on business value, technical value, risk management or strategic fitness.
- Highest ranking items are decomposed into smaller stories during release planning so that they can be completed in future iterations.





THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

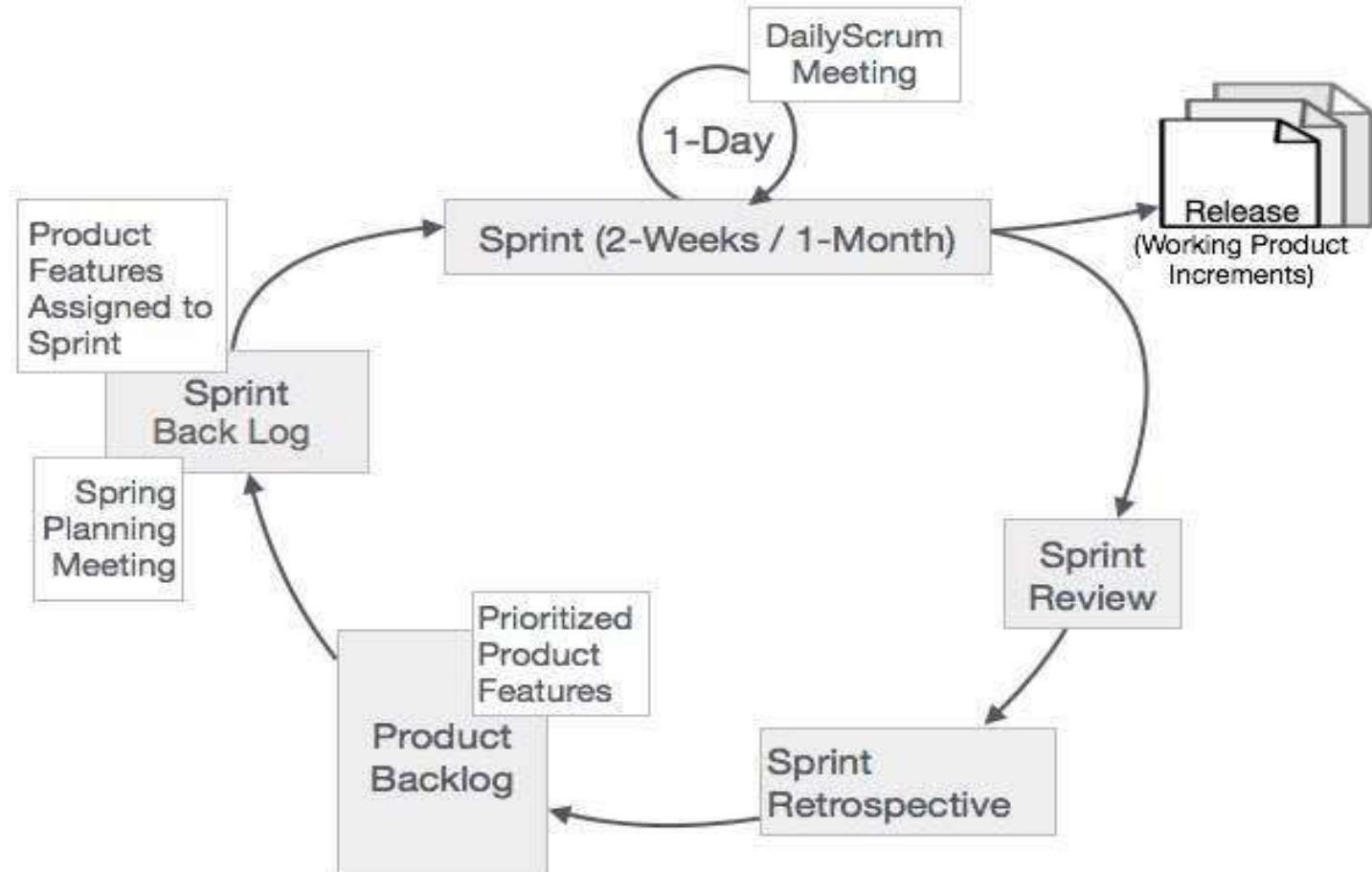
R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Scrum framework

- Scrum is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.
- Scrum is a process framework that has been used to manage complex product development since the early 1990s. Scrum is not a process or a technique for building products; rather, it is a framework within which you can employ various processes and techniques.
- The Scrum framework consists of Scrum Teams and their associated roles, events, artifacts, and rules.



Scrum Process Framework

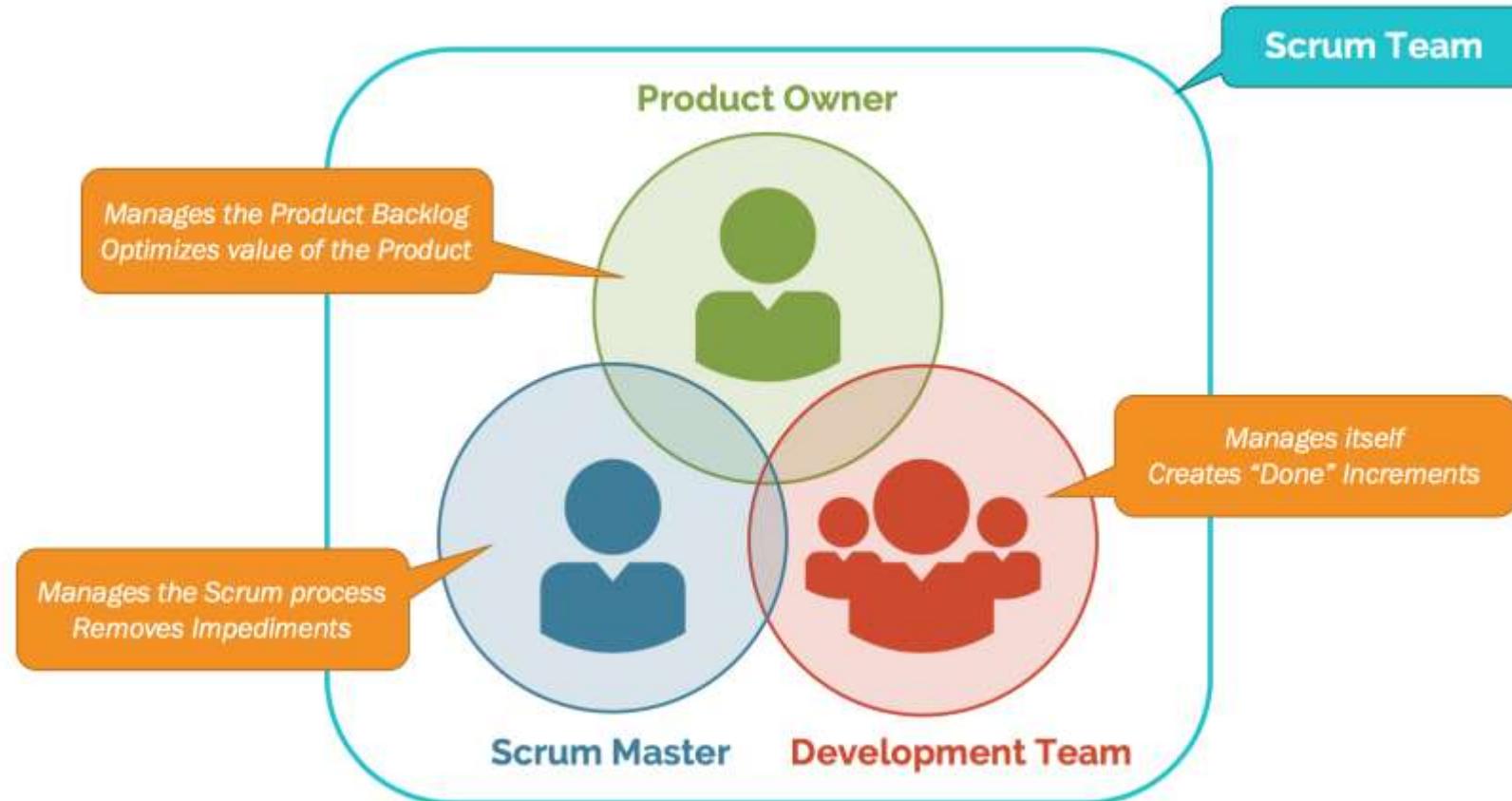


Sprint

- The heart of Scrum is a Sprint, a time-box of two weeks or one month during which a potentially releasable product increment is created. A new Sprint starts immediately after the conclusion of the previous Sprint. Sprints consist of the Sprint planning, daily scrums, the development work, the Sprint review, and the Sprint retrospective.
- In Sprint planning, the work to be performed in the Sprint is planned collaboratively by the Scrum Team.
- The Daily Scrum Meeting is a 15-minute time-boxed event for the Scrum Team to synchronize the activities and create a plan for that day.
- A Sprint Review is held at the end of the Sprint to inspect the Increment and make changes to the Product Backlog, if needed.
- The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning. In this meeting, the Scrum Team is to inspect itself and create a plan for improvements to be enacted during the subsequent Sprint.



Scrum Roles



Scrum Roles

1) ScrumMaster

- The ScrumMaster (sometimes written as the Scrum Master, although the official term has no space after “Scrum”) is the keeper of the scrum process. He/she is responsible for-
- making the process run smoothly
- removing obstacles that impact productivity
- organizing and facilitating the critical meetings



Scrum Roles

2) Product Owner

- The Product Owner is responsible for maximizing the value of the product and the work of the Team. How this is done may vary widely across organizations, Scrum Teams, and individuals.
- The Product Owner is the sole person responsible for managing the Product Backlog. Product Backlog management includes-
 - Expressing Product Backlog items clearly.
 - Ordering the Product Backlog items to best achieve goals and missions.
 - Optimizing the value of the work the Team performs.
 - Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Team will work on further.



Scrum Roles

2) Product Owner

- Ensuring that the Team understands items in the Product Backlog to the level needed.
- The Product Owner may do the above work, or have the Team do it. However, the Product Owner remains accountable for these tasks.
- The Product Owner is one person, not a committee. The Product Owner may represent the desires of a committee in the Product Backlog, but those wanting to change a Product Backlog item's priority must address the Product Owner.
- For the Product Owner to succeed, the entire organization must respect his or her decisions. The Product Owner's decisions are visible in the content and ordering of the Product Backlog. No one is allowed to tell the Team to work from a different set of requirements, and the Team is not allowed to act on what anyone else says. This is ensured by ScrumMaster.



Scrum Roles

3) The Team

- The Team is self-organizing and cross-functional. That means the team comprises of analysts, designers, developers, testers, etc. as appropriate and as relevant to the project.
- Optimal Team size is small enough to remain nimble and large enough to complete significant work within a Sprint. The Team size should be kept in the range from five to nine people, if possible. Fewer than five team members decrease interaction and results in smaller productivity gains. Having more than nine members requires too much coordination.
- The scrum team works together closely, on a daily basis, to ensure the smooth flow of information and the quick resolution of issues. The scrum team delivers product iteratively and incrementally, maximizing opportunities for feedback.



ScrumMaster Services to the Product Owner

- The ScrumMaster serves the Product Owner in several ways, including -
- Finding techniques for effective Product Backlog management.
- Helping the Scrum Team understand the need for clear and concise Product Backlog items.
- Understanding product planning in an empirical environment.
- Ensuring that the Product Owner knows how to arrange the Product Backlog to maximize value.
- Understanding and practicing agility.
- Facilitating Scrum events as needed.



ScrumMaster Services to the Scrum Team

- The ScrumMaster serves the Scrum Team in several ways, including -
- Coaching the Scrum Team in self-organization and cross-functionality.
- Helping the Scrum Team to create high-value products.
- Removing impediments to the Scrum Team's progress.
- Facilitating Scrum events as requested or needed.
- Coaching the Scrum Team in organizational environments in which Scrum is not yet fully adopted and understood.



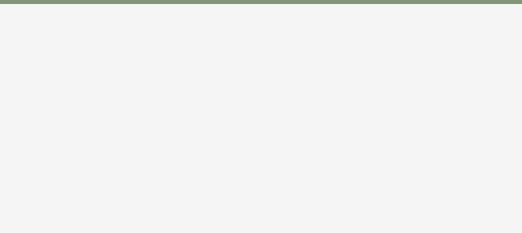
ScrumMaster Services to the Organization

- The ScrumMaster serves the organization in several ways, including-
- Leading and coaching the organization in its Scrum adoption.
- Planning Scrum implementations within the organization.
- Helping employees and stakeholders understand and enact Scrum and empirical product development.
- Causing change that increases the productivity of the Scrum Team.
- Working with other ScrumMasters to increase the effectiveness of the application of Scrum in the organization.





THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Selection of right project management methodology

- There are lots of factors that will impact which project management methodology is right for your project, team, and organization. Here's a quick breakdown of some of the key considerations that can help you decide:
- **Cost and budget:** What sort of budget are you working with? Is there room for that to change if necessary, or is it essential that it stays within these predetermined limits?
- **Team size:** How many people are involved? How many stakeholders? Is your team relatively compact and self-organizing, or more sprawling, with a need for more rigorous delegation?
- **Ability to take risks:** Is this a huge project with a big impact that needs to be carefully managed in order to deliver Very Serious Results? Or is it a smaller-scale project with a bit more room to play around?



There are lots of factors that will impact which project management methodology is right for your project, team, and organization. Here's a quick breakdown of some of the key considerations that can help you decide:

Cost and budget: What sort of budget are you working with? Is there room for that to change if necessary, or is it essential that it stays within these predetermined limits?

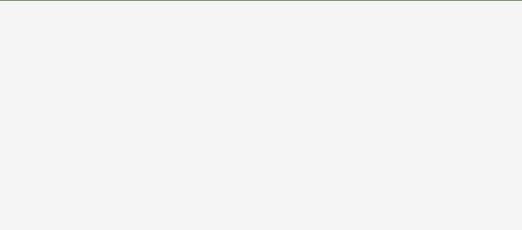
Team size: How many people are involved? How many stakeholders? Is your team relatively compact and self-organizing, or more sprawling, with a need for more rigorous delegation?

Ability to take risks: Is this a huge project with a big impact that needs to be carefully managed in order to deliver Very Serious Results? Or is it a smaller-scale project with a bit more room to play around?

- **Flexibility:** Is there room for the scope of the project to change during the process? What about the finished product?
- **Timeline:** How much time is allotted to deliver on the brief? Do you need a quick turnaround, or is it more important that you have a beautifully finished result, no matter how long it takes?
- **Client/stakeholder collaboration:** How involved does the client/stakeholder need — or want — to be in the process? How involved do you need — or want — them to be?



THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

- **TEXT BOOKS**

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

- **REFERENCES**

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

What is a project management methodology?

- A project management methodology is a set of principles and practices that guide you in organizing your projects to ensure their optimum performance.
- No two projects are exactly the same (even when you're using handy features like project templates to replicate your past successes).
- And when you factor in the different goals, KPIs and production methods of not only different types of teams but also different types of *industries*, it makes sense that there's no one-size-fits-all approach to managing a project.



Selecting the right project management methodology?

- **Cost and budget:** What sort of budget are you working with? Is there room for that to change if necessary, or is it essential that it stays within these predetermined limits?
- **Team size:** How many people are involved? How many stakeholders? Is your team relatively compact and self-organizing, or more sprawling, with a need for more rigorous delegation?
- **Ability to take risks:** Is this a huge project with a big impact that needs to be carefully managed in order to deliver Very Serious Results? Or is it a smaller-scale project with a bit more room to play around?



How do you choose the right project management methodology?

- **Flexibility:** Is there room for the scope of the project to change during the process? What about the finished product?
- **Timeline:** How much time is allotted to deliver on the brief? Do you need a quick turnaround, or is it more important that you have a beautifully finished result, no matter how long it takes?
- **Client/stakeholder collaboration:** How involved does the client/stakeholder need — or want — to be in the process? How involved do you need — or want — them to be?



The project management methodologies list

1. Waterfall methodology
2. Agile methodology
3. Scrum methodology
4. Kanban methodology
5. Scrumban methodology
6. eXtreme programming (XP) methodology
7. Adaptive project framework (APF) methodology
8. Lean methodology
9. Critical path method
10. Critical chain project management



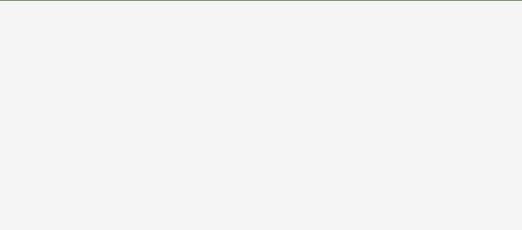
The project management methodologies list

11. New product introduction (NPI)
12. Package enabled reengineering (PER)
13. Outcome mapping
14. Six Sigma
15. PMI's PMBOK
16. PRINCE2 methodology
17. Rapid application development (RAD) methodology





THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

CONTENT OF THE SYLLABUS

- **TEXT BOOKS**

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

- **REFERENCES**

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

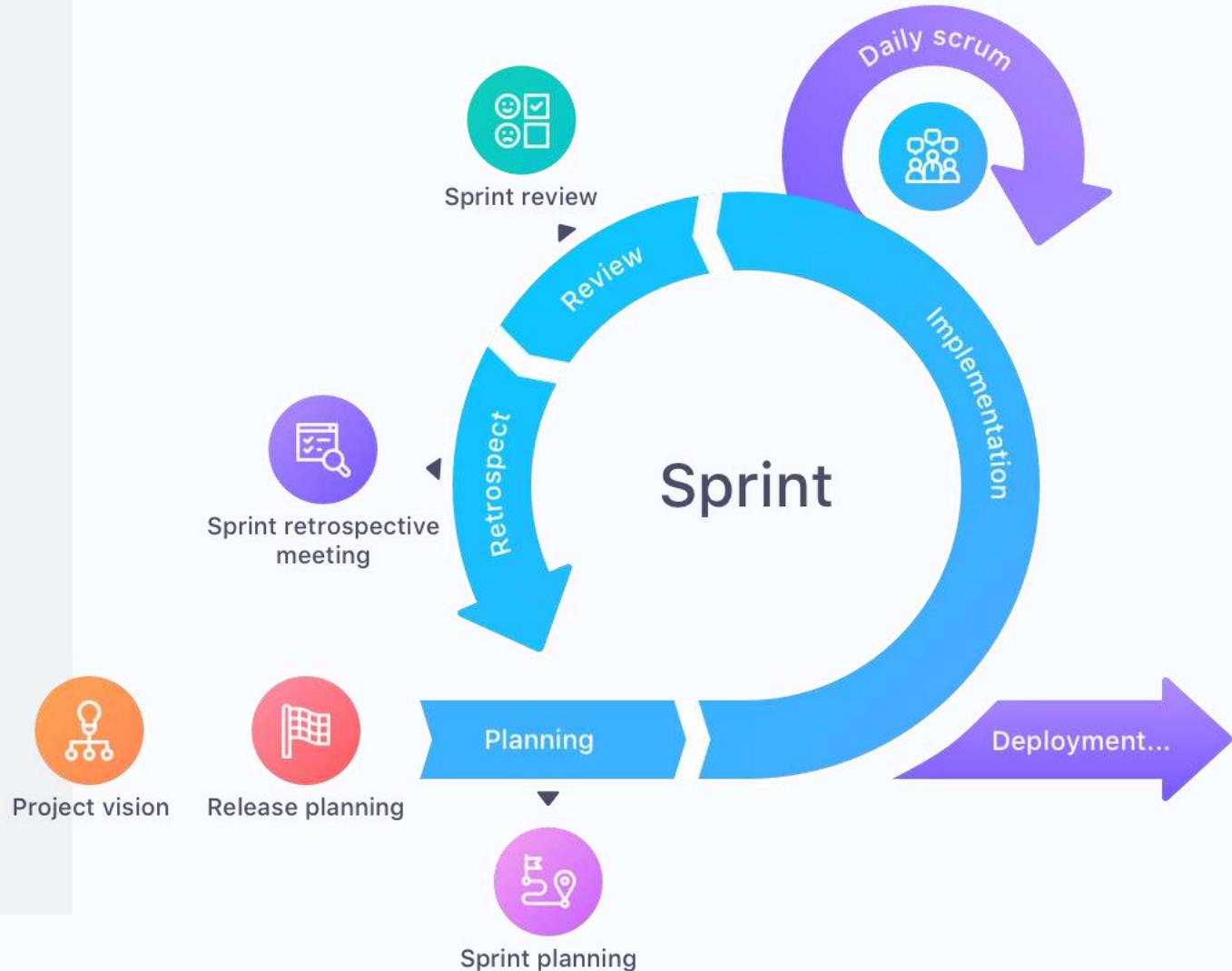
R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Agile Scrum

- **Scrum is a framework** that helps agile teams to work together. Using it, the team members can deliver and sustain the complex product.
- It encourages the team to learn through practice, self-organize while working on the problem. Scrum is a work done through the framework and continuously shipping values to customers.
- **Scrum and agile are not the same** thing because Scrum focused on continuous improvement, which is a core foundation of agile. Scrum framework focuses on ongoing getting work done.



Scrum process

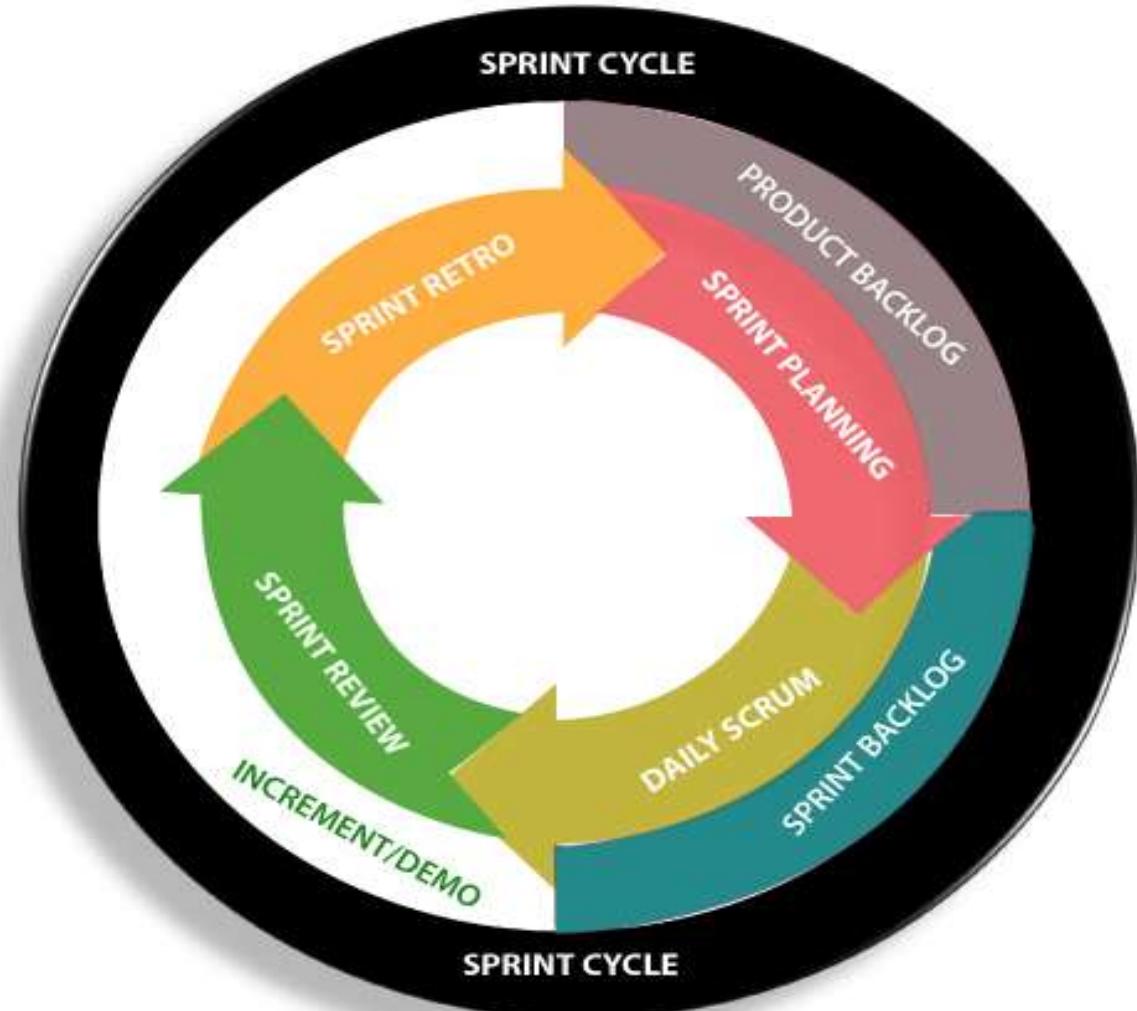


What are sprints?

- With scrum, a product is built in a series of repetition called **sprints**. It breaks down big complex projects into bite-size pieces. It makes projects more manageable, allows teams to ship high quality, work faster, and more frequently. The sprints give them more flexibility to adapt to the changes.
- Sprints are a short, time-boxed period for Scrum team that works to complete a set amount of work. Sprints are the core component of Scrum and agile methodology. The right sprints will help our agile team to ship better software.



What are sprints?



What is sprint plan?

- Sprint plan is an action in Scrum that kicks off the sprint. The primary purpose of sprint plan is to define what can deliver in the sprint. It also focuses on how the work will be achieved. It is done in combination with the whole Scrum team members.
- The sprint is a set of the period where all the work to be done. Before we start the development, we have to set up the sprint. We need to describe how long time is required to achieve the sprint goal and where we are going to start.



Factors affecting Sprint planning

- **The What:** The product owner describes the goal of the sprint and the backlog items which contribute to achieve that goal.
- **The How:** Agile development team plans its necessary work on how to achieve and deliver the sprint goal.
- **The Who:** The product owner defines the goal based on the value that the customers seek. And the developer needs to understand how they can or cannot deliver that goal.
-



Factors affecting Sprint planning

- **The Inputs:** The product backlog provides the list of input stuff that could potentially be part of the current sprint. The team looks over the existing work done in incremental ways.
- **The Outputs:** The critical outcome of sprint planning is to meet described team goal. The product set the goal of sprint and how they will start working towards the goal.
-



What is the product backlog?

- A product backlog is a registered list of work for the development team. It is driven from the roadmap and its requirements.
- The essential task is represented at the top of the product backlog so that the team member knows what to deliver first.
- The developer team doesn't work through the backlog from the product owner's side and product owner doesn't push the work to the developer team.
- The developer team pulls work from the product backlog.

What is a project management methodology?

- A project management methodology is a set of principles and practices that guide you in organizing your projects to ensure their optimum performance.
- No two projects are exactly the same (even when you're using handy features like project templates to replicate your past successes).
- And when you factor in the different goals, KPIs and production methods of not only different types of teams but also different types of *industries*, it makes sense that there's no one-size-fits-all approach to managing a project.



How do you choose the right project management methodology?

- **Cost and budget:** What sort of budget are you working with? Is there room for that to change if necessary, or is it essential that it stays within these predetermined limits?
- **Team size:** How many people are involved? How many stakeholders? Is your team relatively compact and self-organizing, or more sprawling, with a need for more rigorous delegation?
- **Ability to take risks:** Is this a huge project with a big impact that needs to be carefully managed in order to deliver Very Serious Results? Or is it a smaller-scale project with a bit more room to play around?



How do you choose the right project management methodology?

- **Flexibility:** Is there room for the scope of the project to change during the process? What about the finished product?
- **Timeline:** How much time is allotted to deliver on the brief? Do you need a quick turnaround, or is it more important that you have a beautifully finished result, no matter how long it takes?
- **Client/stakeholder collaboration:** How involved does the client/stakeholder need — or want — to be in the process? How involved do you need — or want — them to be?



The project management methodologies list

1. Waterfall methodology
2. Agile methodology
3. Scrum methodology
4. Kanban methodology
5. Scrumban methodology
6. eXtreme programming (XP) methodology
7. Adaptive project framework (APF) methodology
8. Lean methodology
9. Critical path method
10. Critical chain project management



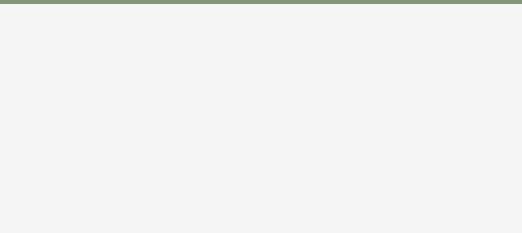
The project management methodologies list

11. New product introduction (NPI)
12. Package enabled reengineering (PER)
13. Outcome mapping
14. Six Sigma
15. PMI's PMBOK
16. PRINCE2 methodology
17. Rapid application development (RAD) methodology





THANK YOU



UNIVERSITY INSTITUTE OF COMPUTING

Agile Methodology

(24CAT-656)

Unit-2- Syllabus

Unit-2	Agile	Lecture Hours:10
Agile Project Management	Project Management introduction, Agile Scrum, Project management methodology, Selection of right project management methodology, Continuous integration and continuous delivery (CI/CD) in agile development.	
Scrum	Scrum framework, Scrum Roles, Agile Kanban, Agile Vs. Scrum. Product Backlog, Scrum Practices, Process flow of Scrum Methodologies,\	
Agile Design	Agile Daily Stand-up, Sprint Review meeting vs Daily Stand-up meeting in Agile, Definition of Done, Agile Design, Retrospective in Agile development.	

• TEXT BOOKS

T1 David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.

T2 Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

T3 Agile Software Development Ecosystems by Jim Highsmith, Addison-Wesley 2002, ISBN 0201760436.

• REFERENCES

R1 Craig Larman, Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

R2 Kevin C. Desouza, Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

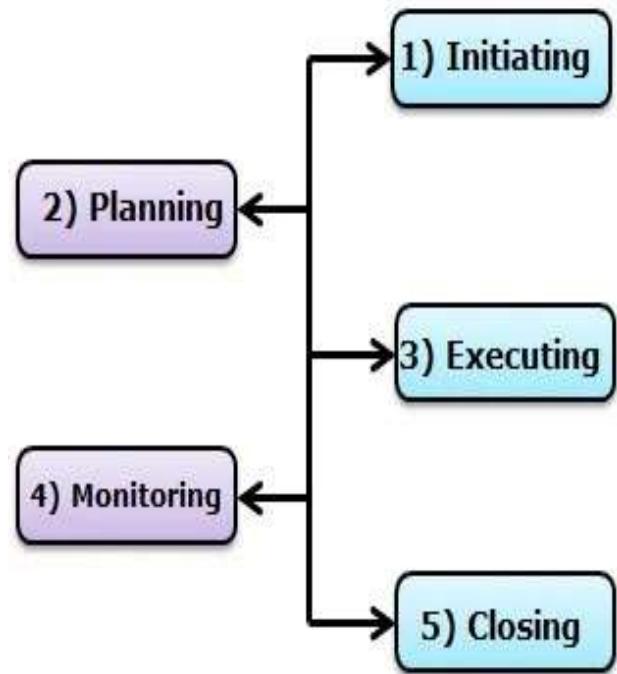
Introduction to Project Management

- Henry Gantt was one of the pioneers of project management who developed the popular Gantt Chart, which is used as a diagram for project management.
- ‘Projects’ are tasks that could be part of our routine lives or any business. Projects could be anything from preparing a meal at home or organizing a vacation tour. And, if it’s business or work-related, developing a website or software, product or a tool, construction of a building can be termed as ‘Projects’.



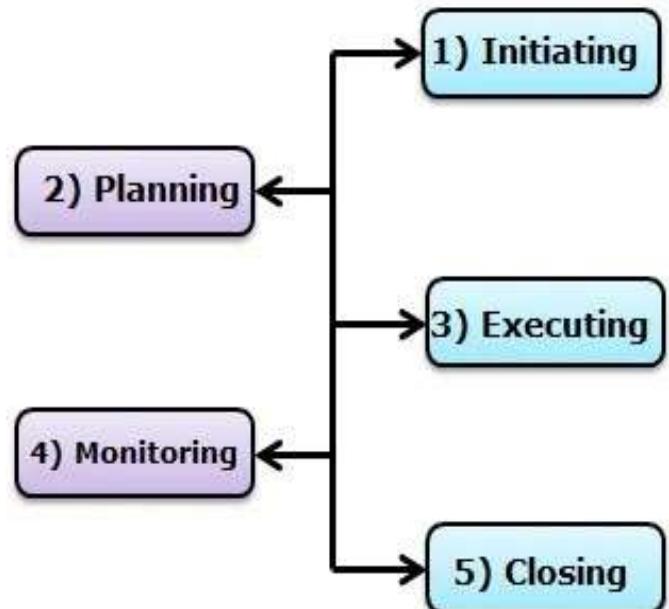
Elements of Project Management

- **Initiating:** The first stage of PM, where formalizing of the project plan, charter and scope are defined so as to determine the cost, resources, timeline, and budgeting for the project.
- **Planning:** The most crucial stage of all, this stage includes strategizing the scope of a project, identifying risks, if any, and developing mitigation plans by creating a set of tasks to administer the project.



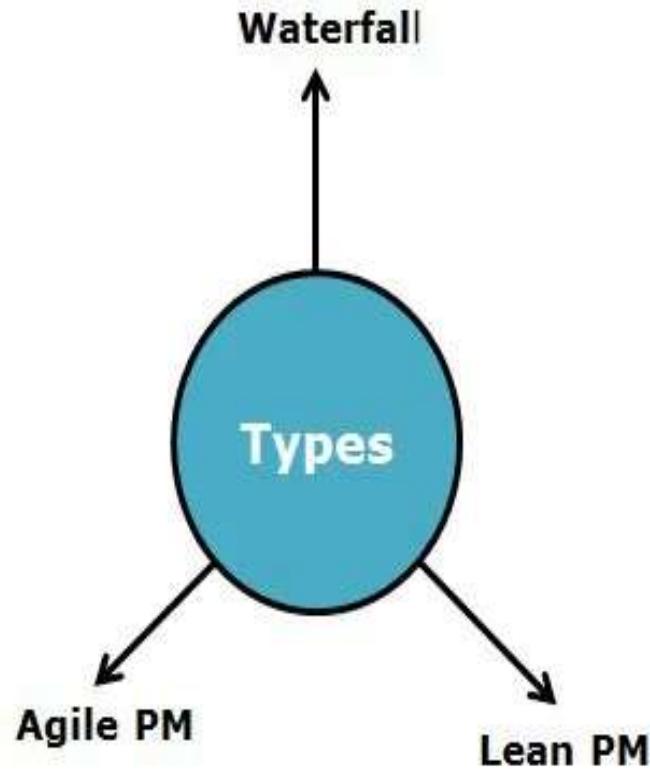
Elements of Project Management

- **Executing:** This is a stage where the project team is ready to launch or administer the project. The tasks are delegated and resumed so as to keep track of the project and efficiently manage the same.
- **Monitoring/Controlling:** At this stage, evaluation of project performance is undertaken by comparing the real-time results to the defined actuals so as to ensure all the goals and deliverables are met.
- **Closing:** The last stage of a project where deliverables are surrendered to the customers, forwarding of documents to the business, the release of resources, and notifying the closure of the project to the stakeholders are carried out.



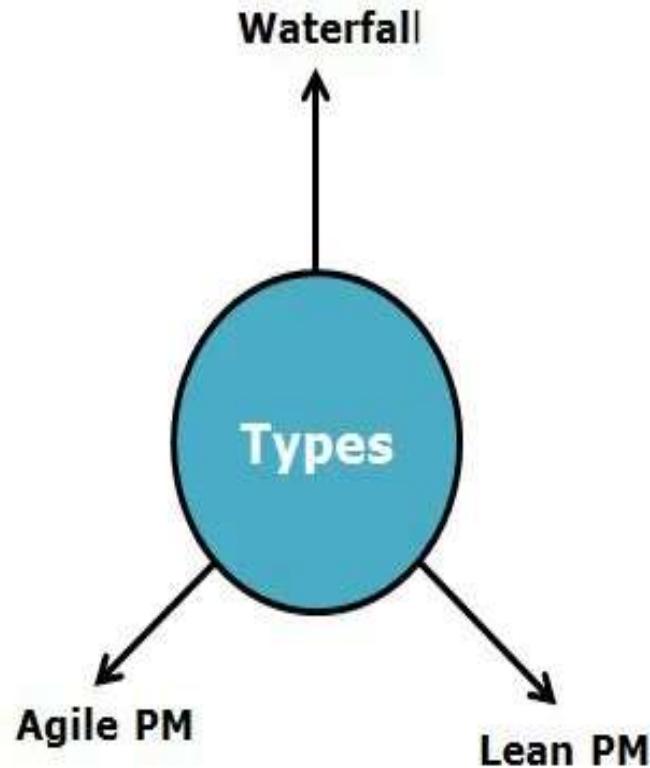
Types of Project Management

- **Waterfall PM:** As the name suggests, this is the conventional type of PM that follows the traditional pattern wherein the next task is resumed only after the completion of the previous task.
- Project timeline and attention to the sequence of tasks are paramount in this type of PM. In this type of PM, the team size goes on expanding as the smaller tasks are finished and the larger ones begin.



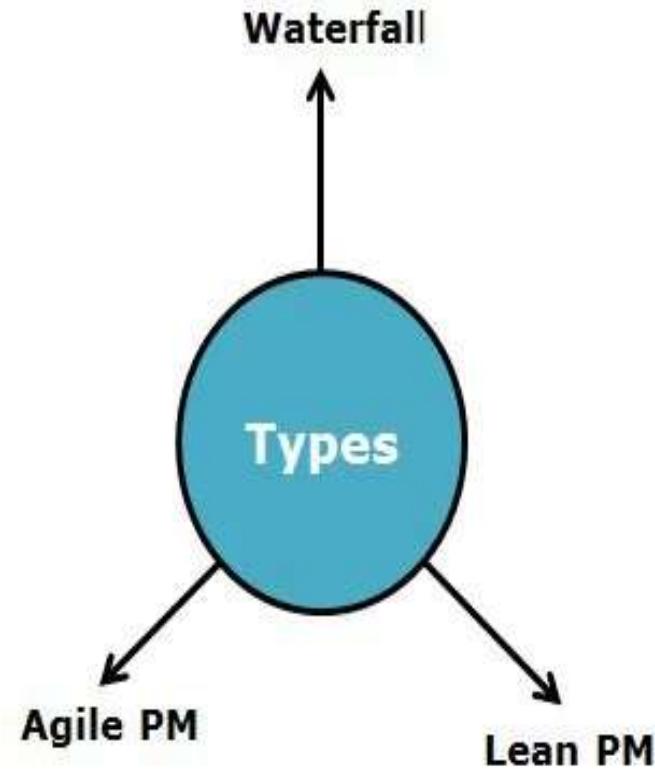
Types of Project Management

- **Agile PM:** The IT sector were the pioneers who use this type of PM. Unlike the conventional one, Agile follows the continual process of improvement through monitoring, continuous follow-up, and interactions with the team members as per the business conditions.
- It helps in identifying and rectification of errors without having to resume the entire process all over again, which is a plus



Types of Project Management

- **Lean PM:** Lean PM is based on the principle of avoiding waste of time and resources. It practices the methodology of creating more value from less, which is primarily based on the Japanese way of functioning.
- The lean way of PM focuses on adding value by the elimination of everything else that fails to add value!



Role of Project Manager

- The Project Manager is the ‘captain of the ship’ who is leading the project team, plans and executes a project.
- The project manager is required to ensure whether or not the project is in alignment with the customer’s vision and quality standards.
- The success or failure of the project is bestowed on the shoulders of the Project Manager.



Role of Project Manager

- Identifying and defining the project scope, timeline, resources, and developing the budget.
- Formalizing the identified factors and putting forth the documents, preparing schedules and charts.
- Monitoring, evaluating, and reporting the progress to the stakeholders
- Conducting risk analysis, managing risks, and developing mitigation plans.



Role of Project Manager

- Communicating and negotiating with the stakeholders
- Maintaining the quality standards of a project.
- Monitoring and managing the results as per the defined actuals.
- Closing the project by analyzing the results and communicating the same to the stakeholders.





THANK YOU

