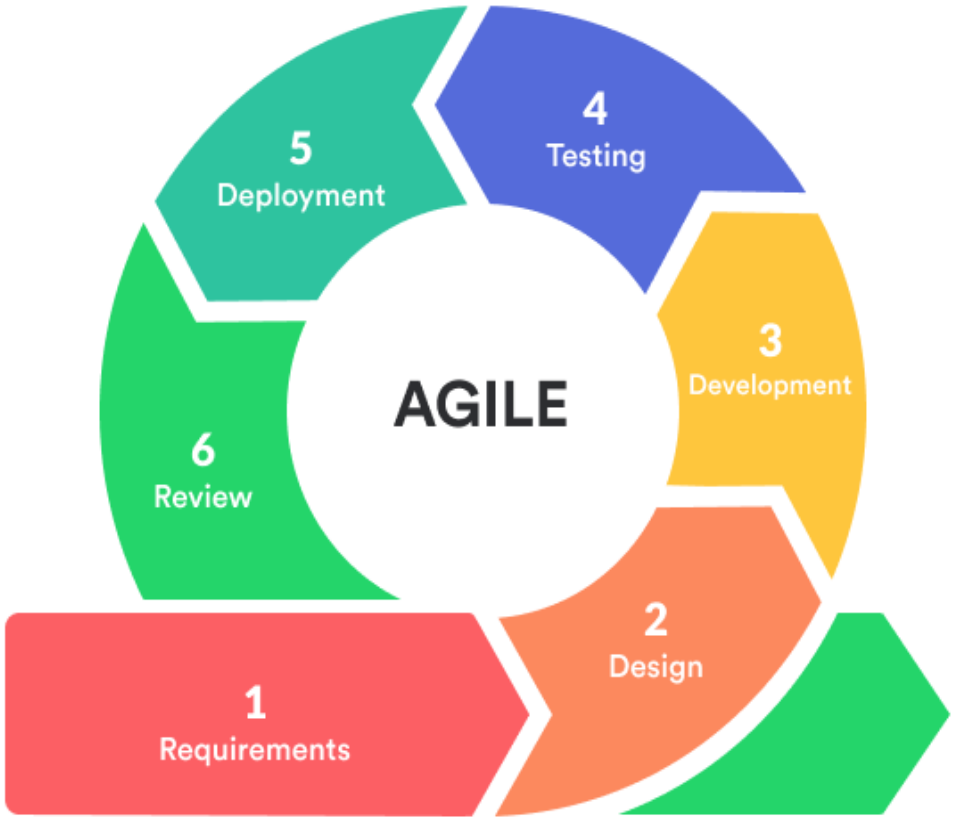


AGILE DEVELOPMENT

GROUP 9

ALANKAR SAXENA 1803211005

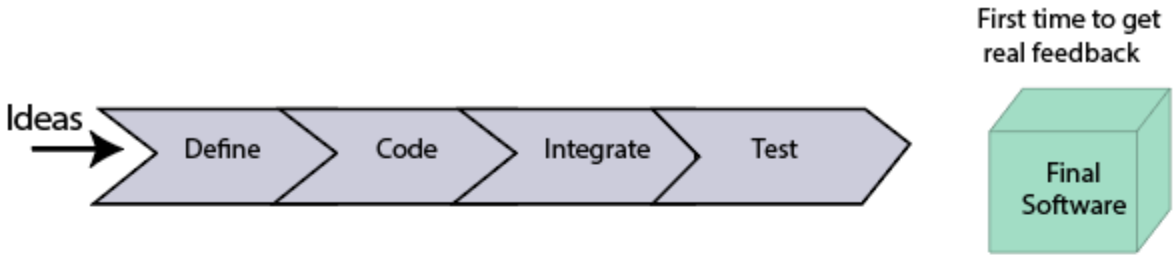
PRAGATI GUPTA 1803213111

ATUL KUMAR 1803210036

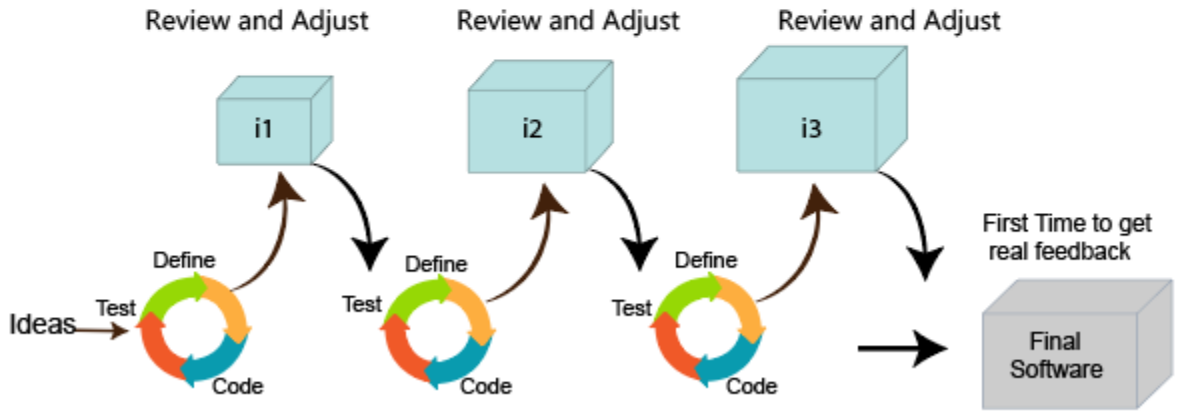
SUBMITTED TO : Harivans Pratap Singh

❖ What is Agile Methodology?

An agile methodology is an iterative approach to software development. Each iteration of agile methodology takes a short time interval of 1 to 4 weeks. The agile development process is aligned to deliver the changing business requirement. It distributes the software with faster and fewer changes.



Traditional Method



Agile Method

The single-phase software development takes 6 to 18 months. In single-phase development, all the requirement gathering and risks management factors are predicted initially.

The agile software development process frequently takes the feedback of workable product. The workable product is delivered within 1 to 4 weeks of iteration.

❖ Need of Agile

1. Customer satisfaction is rapid, continuous development and delivery of useful software.
2. Customer, Developer, and Product Owner interact regularly to emphasize rather than processes and tools.
3. Product is developed fast and frequently delivered (weeks rather than months.)
4. A face-to-face conversation is the best form of communication.
5. It continuously gave attention to technical excellence and good design.
6. Daily and close cooperation between business people and developers.
7. Regular adaptation to changing circumstances.
8. Even late changes in requirements are welcomed.

❖ Agile Manifesto

- **Customer Satisfaction:** Manifesto provides high priority to satisfy the costumer's requirements. This is done through early and continuous delivery of valuable software.
- **Welcome Change:** Making changes during software development is common and inevitable. Every changing requirement should be welcome, even in the late development phase. Agile process works to increase the customers' competitive advantage.
- **Deliver the Working Software:** Deliver the working software frequently, ranging from a few weeks to a few months with considering the shortest time period.
- **Collaboration:** Business people (Scrum Master and Project Owner) and developers must work together during the entire life of a project development phase.
- **Motivation:** Projects should be built around motivated team members. Provide such environment that supports individual team members and trust them. It makes them feel responsible for getting the job done thoroughly.
- **Face-to-face Conversation:** Face-to-face conversation between Scrum Master and development team and between the Scrum Master and customers for the most efficient and effective method of conveying information to and within a development team.
- **Measure the Progress as per the Working Software:** The working software is the key and primary measure of the progress.
- **Maintain Constant Pace:** The aim of agile development is sustainable development. All the businesses and users should be able to maintain a constant pace with the project.
- **Monitoring:** Pay regular attention to technical excellence and good design to maximize agility.
- **Simplicity:** Keep things simple and use simple terms to measure the work that is not completed.
- **Self-organized Teams:** The Agile team should be self-organized. They should not be depending heavily on other teams because the best architectures, requirements, and designs emerge from self-organized teams.
- **Review the Work Regularly:** The work should be reviewed at regular intervals, so that the team can reflect on how to become more productive and adjust its behavior accordingly.

❖ Roles in Agile

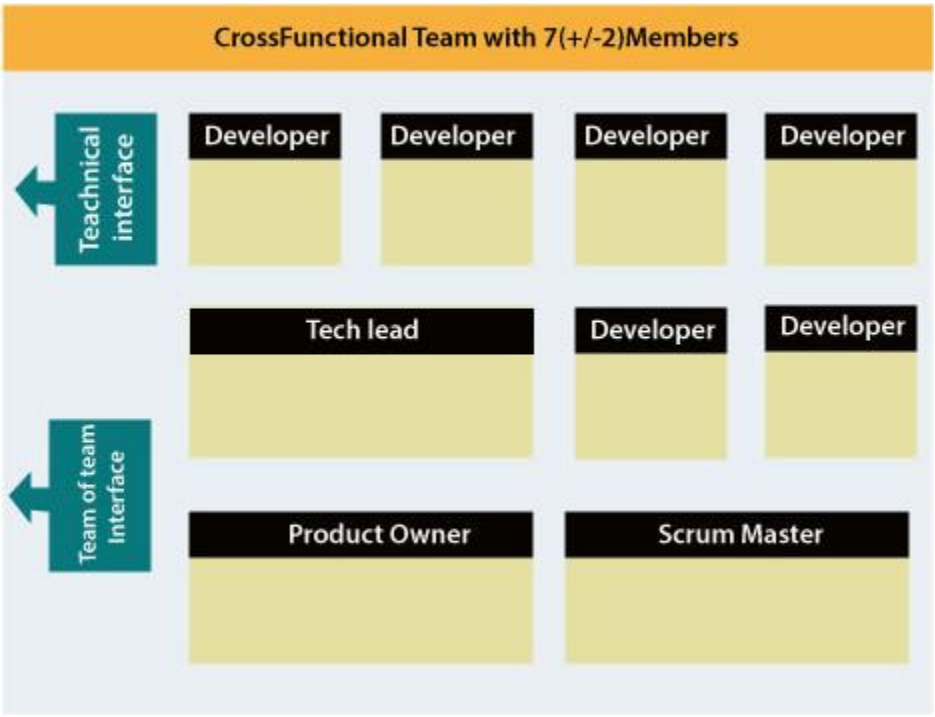
There are three different roles in Agile methodology. These are :

- **Scrum Master :** The Scrum Master is a team leader and facility provider who helps the team member to follow agile practices, so that the team member meets their commitments and customers' requirements. The scrum master plays the following responsibilities :
  - They enable the close co-operation between all the roles and functions.
  - They remove all the blocks which occur.
  - They safeguard the team from any disturbances.
  - They work with the organization to track the progress and processes of the company.
  - They ensure that Agile Inspect & Adapt processes are leveraged correctly which includes :
    - Planned meetings
    - Daily stand-ups
    - Demo
    - Review
    - Retrospective meetings, and
    - Facilitate team meetings and decision-making process.
- **Product Owner :** The Product Owner is one who runs the product from a business perspective. The Product Owner plays the following responsibilities :
  - He defines the requirements and prioritizes their values.
  - He sets the release date and contents.
  - He takes an active role in iteration and releasing planning meetings.
  - He ensures that the team is working on the most valued requirement.
  - He represents the voice of the customer.
  - He accepts the user stories that meet the definition of done and defined acceptance criteria.

❖ Cross – functional Team

Every agile team contains self-sufficient team with 5 to 9 team members. The average experience of each member ranges from 6 to 10 years. The agile team contains **3 to 4 developers, 1 tester, 1 technical lead, 1 scrum master and 1 product owner**.

The Scrum master and Product owner are considered as a part of Team Interface, on the other hand remaining members are the part of Technical Interface.



❖ How an Agile Team plan its Work?

An Agile team works in iterations to deliver user stories where each iteration is of 10 to 15 days. Each user story is planned based on its backlog prioritization and size. The team uses its capacity − how many hours are available with team to work on tasks − to decide how much scope they have to plan.

- **Point :** A Point defines how much a team can commit. A point usually refers to 8 hours. Each story is estimated in points.
- **Capacity :** Capacity defines how much an individual can commit. Capacity is estimated in hours.

❖ How requirement is Done

The Agile team decides the meaning of task done. There may be different criteria for it:

- When the entire task (development, testing) are completed.
- When all the acceptance tests are running and are passed.
- When no defects found.
- Product owner has accepted the requirement.
- When the software product is delivered to the end user.

❖ What is User Story

A user story is a requirement which defines what is required by the user as functionality. A user story can be in two forms −

- As a <User Role> I want <Functionality> so that <Business Value>
- In order to <Business value> as a <User Role> I want <Functionality>

During release planning, a rough estimate is given to a user story using relative scale as points. During iteration planning, the story is broken down into tasks.

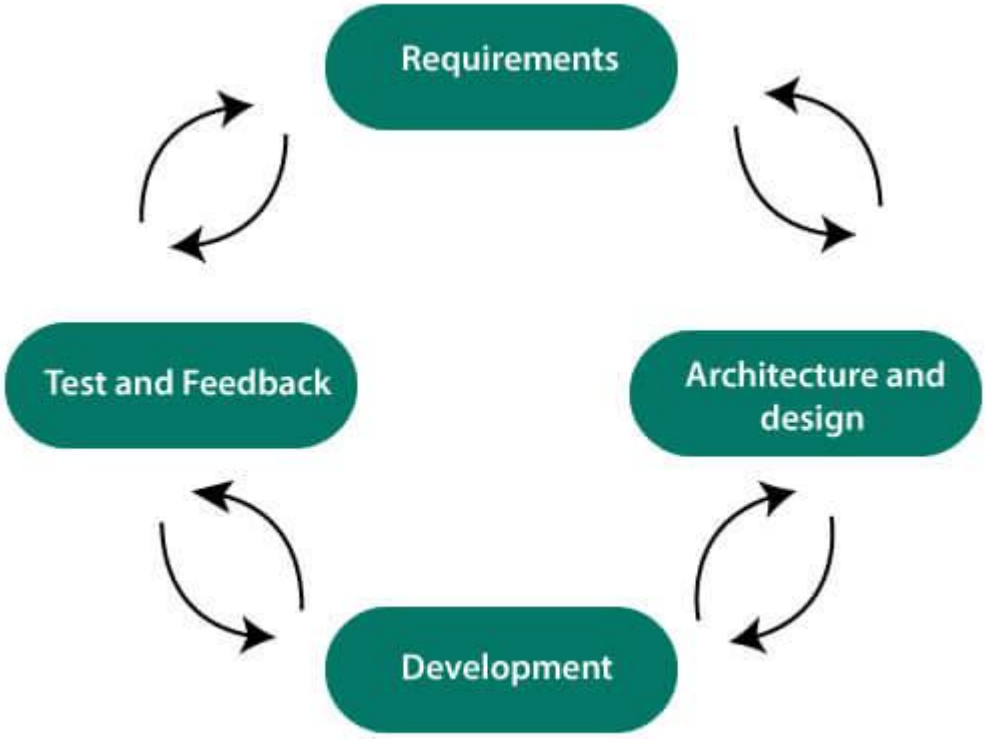
❖ What is Product Backlog

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 it 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**
  - 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.

❖ Agile Software Development Life Cycle (SDLC)

Agile Software Development Life Cycle (SDLC) is the combination of both iterative and incremental process models. It focuses on process adaptability and customer satisfaction by rapid delivery of working software product. Agile SDLC breaks down the product into small incremental builds. These builds are provided into iterations.

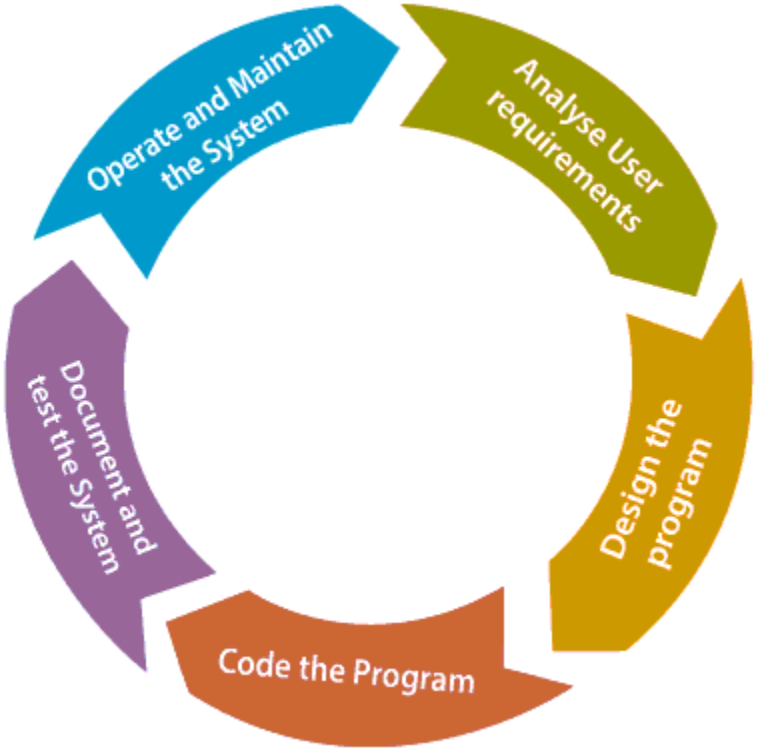


In the agile SDLC development process, the customer is able to see the result and understand whether he/she is satisfied with it or not. This is one of the advantages of the agile SDLC model. One of its disadvantages is the absence of defined requirements so, it is difficult to estimate the resources and development cost.

Each iteration of agile SDLC consists of cross-functional teams working on various phases:

- **Requirements gathering and analysis :** In this phase, you must define the requirements. You should explain business opportunities and plan the time and effort needed to build the project. Based on this information, you can evaluate technical and economic feasibility.
- **Design the requirements :** When you have identified the project, work with stakeholders to define requirements. You can use the user flow diagram or the high-level UML diagram to show the work of new features and show how it will apply to your existing system.
- **Construction / Iteration :** When the team defines the requirements, the work begins. The designers and developers start working on their project. The aims of designers and developers deploy the working product within the estimated time. The product will go into various stages of improvement, so it includes simple, minimal functionality.

- **Deployment :** In this phase, the team issues a product for the user's work environment.
- **Testing :** In this phase, the Quality Assurance team examine the product's performance and look for the bug.
- **Feedback :** After releasing of the product, the last step is to feedback it. In this step, the team receives feedback about the product and works through the feedback.



Agile SDLC Process Flow

1. **Concept:** Project are imagined and prioritized.
2. **Inception:** Team members are created, funding is put in place, and basic environments and requirements are discussed.
3. **Iteration/Constriction:** The software development team works to deliver working software. It is based on requirement and feedback.
4. **Release:** Perform quality assurance (QA) testing, provides internal and external training, documentation development, and final version of iteration into the product.
5. **Production:** It is ongoing support of the software.

Advantages of Agile SDLC

1. Project is divided into short and transparent iterations.
2. It has a flexible change process.
3. It minimizes the risk of software development.
4. Quick release of the first product version.
5. The correctness of functional requirement is implemented into the development process.
6. Customer can see the result and understand whether he/she is satisfied with it or not.

Disadvantages of Agile SDLC

1. The development team should be highly professional and client-oriented.
2. New requirement may be a conflict with the existing architecture.
3. With further correction and change, there may be chances that the project will cross the expected time.
4. There may be difficult to estimate the final coast of the project due to constant iteration.
5. A defined requirement is absent.

❖ Agile Project Management

Agile project management is an interactive approach to manage software development. The agile project management focuses on continuous releases and covers customer feedback with every iteration.

Traditionally the agile project management is classified into two frameworks: scrum and kanban. The scrum framework focused fixed-length project iterations, whereas kanban framework focused on continuous releases. After competition of project first iteration (or steps) project management activity immediately moves on to the next.

History of Agile Project Management

Agile project management is rapidly rising in the 21st century. It is used for software development projects and other IT initiatives. However, from the mid-20th century, the concept of continuous development has taken various forms. For example, there was James Martin's **Rapid Iterative Production Prototyping (RIPP)**, an approach that served as the premise for the 1991 book **Rapid Application Development (RAD)**.

The agile project management framework which has emerged in most recent years is known as Scrum. This methodology features works on the development team to create a product backlog. It also creates a prioritized list of the features, functionalities, and fixes required to deliver a successful software system. The scrum team offers the pieces of a task in rapid increments.

How Agile Project Management Works

The agile project management calls for teams to regularly evaluate cost and time as they move through their work. They use velocity, burnup and burndown charts to measure their work, rather than Gantt charts and project milestones to track progress.

The agile team practices to continuous development and continuous integration using technology that automates steps to speed up the release and use of products.

The presence and participation of the project manager are not required in agile project management. Although the presence of the project manager is essential for success under the traditional (waterfall model) project delivery. The role of the project manager is to distribute task among team members. However, the project manager is not obsolete in

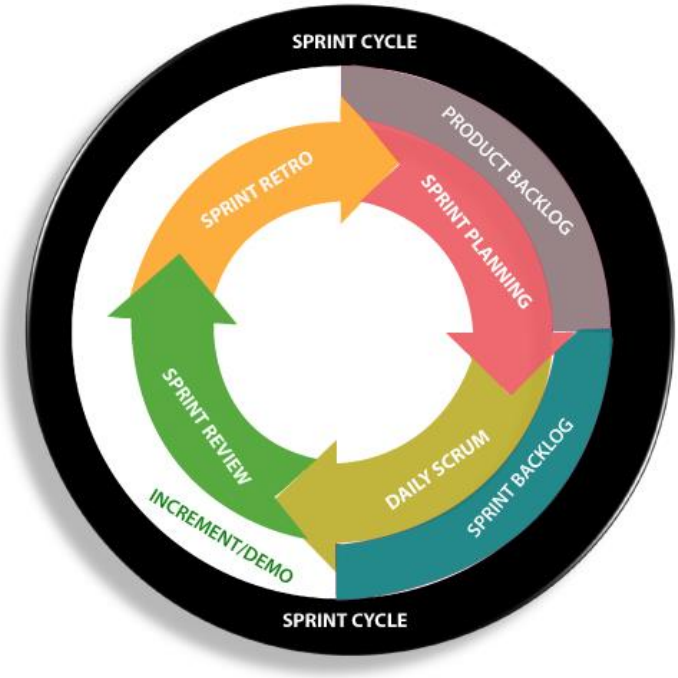
agile project management, and many organizations use them in a large, more complex project. The organization mostly places them in the project coordinator role.
Agile Project Management demands that team members know how to work in this new agile methodology. The team member must be able to coordinate with each other, as well as with users.

❖ What is 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. Scum is a work done through the framework and continuously shipping values to customers.
It is the most frequent software that is used by the development team. Its principle and lessons can be applied to all kinds of teamwork. Its policy and experiences is a reason of popularity of Scrum framework. The Scrum describes a set of tools, meetings, and roles that help the teams structure. It also manages the work done by the team

❖ 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.



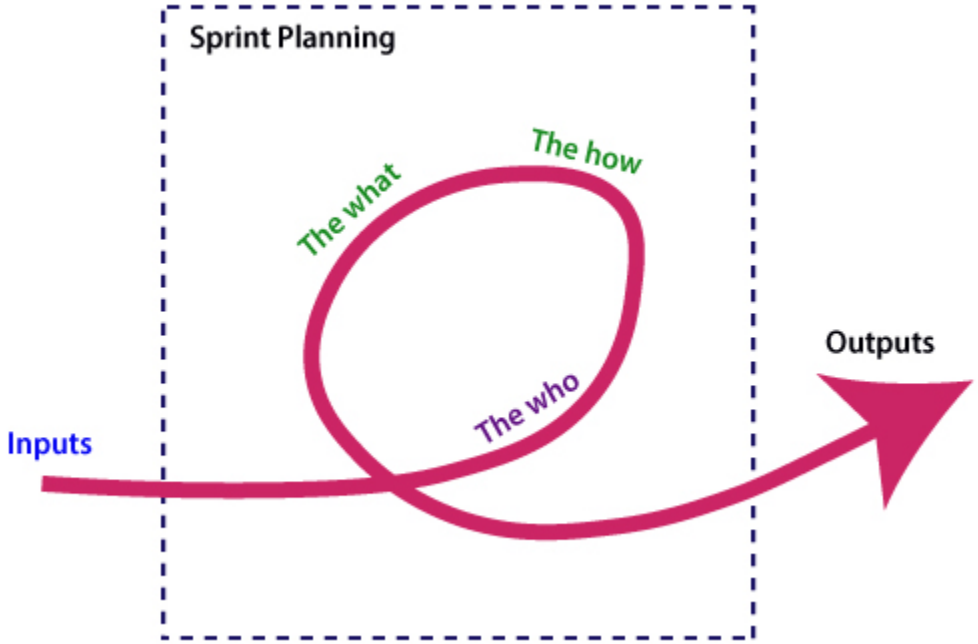
❖ What is Sprint Plan?

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.
- **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 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 Kanban?

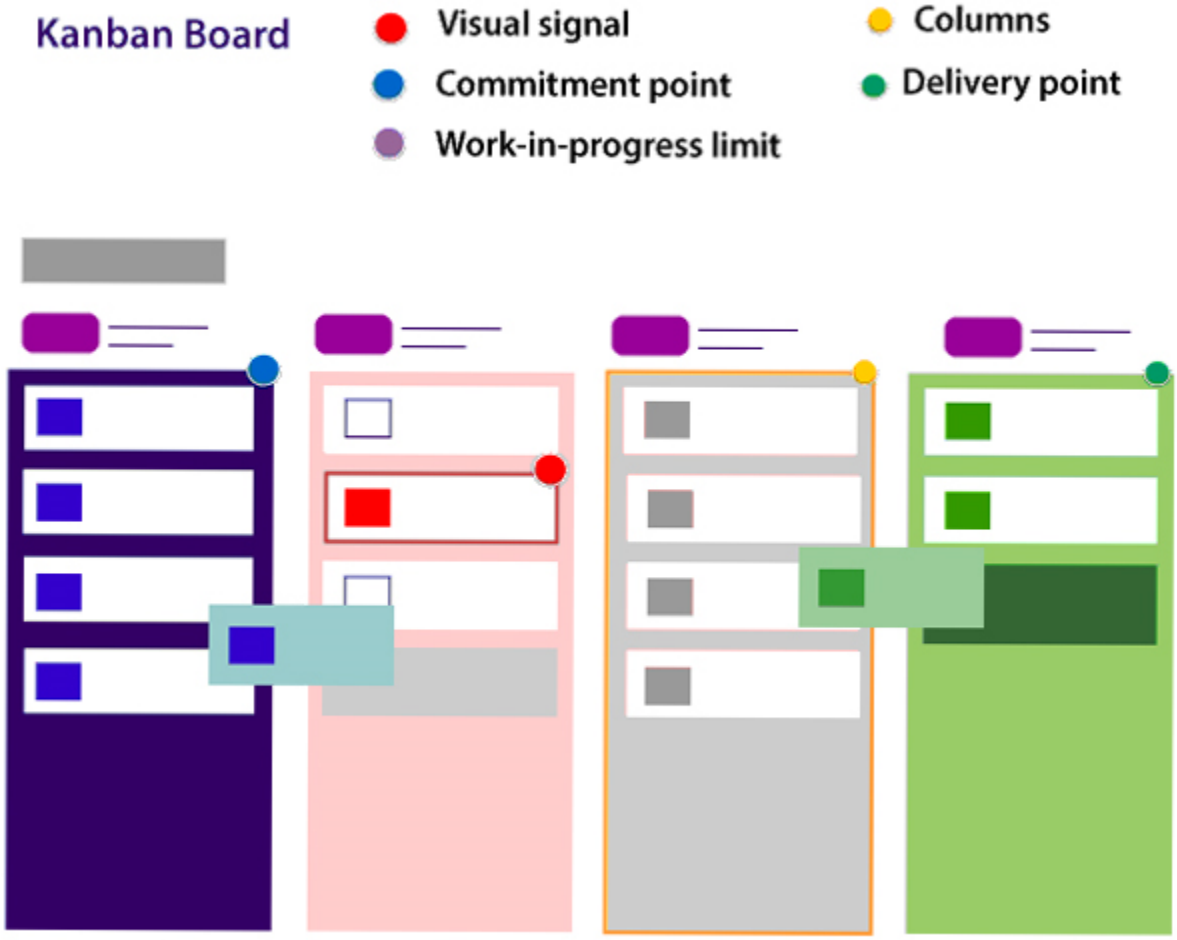
Kanban is a popular framework which is used to implement agile software development. It takes real time communication of capacity and complete transparency of work. The work items are represented in a kanban board visually, allowing team members to see the state of every piece of work at any time.

Boards

The kanban board is the agile project management tool that designed the necessary visualized work, limited work-in-progress, and maximizes flow (or efficiency). It uses cards, columns, and provides continuous improvement to help technology and service teams who commit the right amount of work and get it done.

Elements of Kanban board

A person called David Anderson divides the kanban board into five different components. These are Visual signals, columns, work-in-progress limits, a commitment point, and a delivery point.



1. **Visual Signals:** The kanban board is a visual card (stickies, tickets, or otherwise). Kanban team write their projects and work items onto cards, usually per person each card. For agile teams, each card could encapsulate into one user story. Once the board completed, this visual team helps team members and stock members quickly to understand what the team is working.
2. **Columns:** The column represents the specific activities that compose a "workflow" together. The card flows through a workflow until its completion. The workflow may be a simple as "To Do," "In Progress," "Complete," or much more complicated.
3. **Work in progress (WIP) Limits:** The work in progress limits are the maximum number of cards which can be in one column. This is at any given time. It gives the alert signal that you committed too much work.
4. **Commitment point:** Kanban teams also maintain a backlog for their board. This is where the customers and team member put ideas for projects that the team can pick up. The team members pick up plans when they are ready. The committed point is a movement where the design is picked up by the team, and work starts on the project.
5. **Delivery point:** It is the end point of a kanban team's workflow. Mostly the delivery point for every team is when the product and services are handed to the customer.

❖ Kanban v/s Scrum Board

Kanban	Scrum
Kanban is an ongoing process.	Scrum sprints have a start and stop dates
Kanban has no formal roles.	Role is clearly defined of each team in the scrum (product owner, development team, and scrum master). Both teams are self-organized.
A kanban board is used throughout the lifecycle of a project	Scrum board is cleared and recycled after each sprint.
This board is more flexible with regards to tasks and timing. Its task can be reprioritized, reassigned, or updated as needed.	This board has the number of tasks and a strict deadline to complete them.

❖ Difference between Agile and Scrum (Agile v/s Scrum)

Agile is an iterative approach of software development methodology using short iterations of 1 to 4 weeks. Due to the agile methodology, the development process is aligned to deliver the changing business requirement. Using Agile methodology, the software is distributed with faster and fewer changes.

Scrum is a framework of agile that helps agile teams to work together. Using it, the team members development, deliver and sustain the complex product. It encourages the team to learn through practice, self-organize while working on the problem. Scum is a work done through the framework and continuously shipping values to customers.

Agile	Scrum
1. Agile is an **iterative and incremental approach** to software development methodology.	1. Scrum is a framework of agile methodology. In which **incremental builds** are delivered to end user in every two to three weeks.
2. In this approach, the **leadership** plays an important role.	2. Scrum's team is **self-organized**, cross-functional team.
3. Agile software development is highly suitable for the **medium or large project**.	3. Scrum is used in the project where the requirement **rapidly** changes.
4. **Flexibility** is the most significant advantage of agile as it quickly reacts to changes.	4. A compared to agile it is more **rigid**. So that there are no chances of frequent change.
5. Agile involves **face-to-face communication** and collaboration between the members of various cross-functional teams.	5. In **daily stand up meeting** the teamwork is achieved with a fixed role assigned to team members, scrum master, and product owner.
6. Agile development needs **frequent delivery** to the end user for their feedback.	6. **No need to change many more** while implementing scrum process.
7. In this development, each step like requirements, analysis, design, are **continually monitored** during the lifecycle.	7. In this process, a **build is delivered** after each sprint to the client for their feedback.
8. The **project leader** takes cares of all the tasks in the agile method.	8. After every sprint a demonstration of functionality is provided. So that the **regular feedback** can be taken before next sprint.
9. End-user may give their **feedback during the development** process. So, the end product will be more useful.	9. There is no team leader, so the **entire team handles the issues** or problems.

10. **Delivery and update** of the software are taking place regularly. 11. Design and execution should be kept **simple**. 12. The priority of agile development is always to satisfy the customer by providing **continuous delivery** of valuable software. 13. Working software is the most **fundamental measure** of progress. 14. It is best to have **face-to-face communication** to get as close to the project goal as possible.	10. When the team completes the **current sprint activity**, then the next sprint is planned. 11. Design and execution can be **innovative and experimental**. 12. The **daily sprint meeting** is organized to review the feedback to decide the future progress of the project. 13. Working software is **not a fundamental measure**. 14. The target of the Scrum team is to deliver **maximum business value**.

❖ Agile Daily Stand Up

Agile daily stand-up is termed as the day-to-day status meeting on the project of the members of the agile team. The daily meeting of the agile team discussed the forum for regular updates as well as the problems of team members. It focuses on addressing the issues and tries to solve the issues quickly. The daily stand-up is the regular practice, no matter how an agile team is established regardless of its office location.

What is Daily Stand Up?

The daily stand-up is a daily status meeting of the agile team member. This meeting roughly takes 12 to 18 minutes (an average of 15 minutes).

Each member of the team has to answer three important questions

1. What he/she did yesterday?
2. What he/she will do today?
3. The problem he/she is facing . . . He/she blocked due to. . .

The daily stand-up is done for a day-to-day status update. The meeting of team members with the product owner can be scheduled at different time. The participants in the stand-up meetings only stand instead of sitting so that the meetings get finished quickly.

Importance of Daily Stand Up

The importance of having a daily stand-up in agile are as follows:

- The team can evaluate the progress report daily.
- The team member discusses all the progress and the commitments he/she made for the day.

- The members can also see whether they can deliver the project as per the iteration plan or not.
- Stand-up provides visibility to the team on any delay that occurs due to some obstacles.

Who attends a Stand Up?

- The project owner, scrum master, and the delivery team should attend the stand-up regularly.
- Customers and Stakeholders are encouraged to participate in the meeting, and they act as an observer. However, they are not supposed to participate in stand-ups.
- The responsibility of scrum master is to take note each team member's queries and the problems they are facing.

❖ Agile Definition of Done

Agile Definition of **done** is defined into three different stages called User Story (Requirement), Iteration, and product Release. These are given below:

User Story (requirement)

A user story is a requirement which is formulated into few sentences. The user requirement is the everyday language of user. This user story should be completed within iteration. The user story is done when

- All the related code and documentation have been checked-in.
- The product passed all the processes of unit test.
- All the processes of the acceptance test case have been moved.
- The product owner must have accepted the story.
- The help text (documentation) is written.

Iteration

An iteration is a time-based collection of a user story. It works on the defected product and accepted within the release of a product. Iteration is defined at the time of the iteration planning meeting and completed within the iteration demo and review meeting. The iteration is also known as a sprint. The repetition is required when:

- Performance of the product has been tested.
- Product backup is complete.
- User requirement has been accepted or moved for the next iteration.
- Defected product has been fixed or postponed for the next iteration.

Release

The product release is a major occasion that represents an internal and external delivery of work. It also tests the version of the product or system. The product release is done when:

- The system is stress tested.
- Performance is high.
- Contain the security validation in the product.
- Disaster recovery plan is tested.

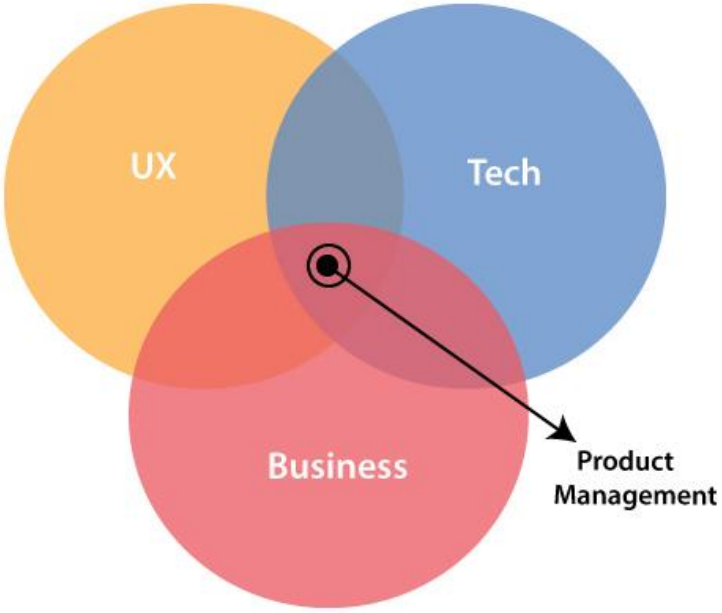
❖ Product Management

Product management is the organizational function which guides every step to product lifecycle. The product lifecycle starts from development to positioning and pricing. Its focus on the product and its customer first and foremost.

The product entirely focuses on the customer. Product teams routinely ship, better designed, and higher performing product. The product team members daily work with product managers and have interviewed dozens of their role and responsibility. In product management, there is no such one way to apply any principle. Every product has its own goals and challenges which require a unique and customized approach to product management.

Factors affecting Product Management

- **Business:** Product management helps teams to achieve their business objective by minimizing the communication gap between product developments, design, the customer, and the company.
- **User Experience:** Product management concentrate on the user experience (UX) that represents the customer within an organization. Better UX is focus manifests itself.
- **Technology:** Product management is a day to day activity in the engineering department. The accurate understanding of computer science is paramount.



❖ What is Agile Product Management

Agile product management is about guiding software development, product management through multiple iterations. As agile programs are more fluid than traditional approaches so that agile product management is a more flexible approach.

Agile product managers are more integrated towards technology team than business teams. The product management is supported by the management team and Product Marketing Managers to round out the product discipline. The product manager work over marketing data and business objective.

❖ Agile Product Requirement

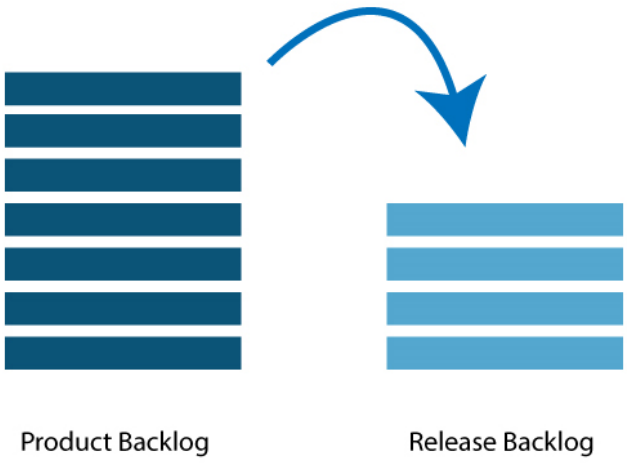
Building a high and new product requires research and comprehensive planning. But a question arises from where to start? The product manager generally begins with the product requirements document (PRD).

A product requirement document defines the product that we are building. Product requirement outlines the product's ambition, its features, functionalities, and behavior.

After collecting all the product requirements, product manager shared it with stakeholders - business and technical teams. They help in building, launch, or market the product.

❖ Agile Release Planning

The primary purpose of release planning is to make a plan to deliver an increment to the product. It is done in the interval of every 2 to 3 months.



Who is Involved in releasing the Plan?

Following person are involved in product releasing plan- Scrum Master, Product Owner, Agile Development Team, Stakeholders.

- **Scrum Master:** The Scrum Master is a team leader and facility provider who helps the team member to follow agile practices so that they can meet their commitments and customers requirements.
- **Product Owner:** The Product Owner is one who runs the product from a business perspective. He defines the requirements and prioritizes their values.
- **Agile Development Team:** Agile development team provides the judgment on the technical feasibilities or any dependencies.
- **Stakeholders:** Stakeholders are the customers, subject matter, program manager act as advisers in decisions which are made around the release planning.

Prerequisites of Planning

The prerequisites of release planning are as follows:

- A Product Owner manages the ranked product backlog. While releasing the product, the Product owner feels to include five to ten features at the period of product release.
- High- level vision
- Market and Business objective
- Team's input according to capabilities, known velocity, or about any technical challenge.
- Acknowledged about the new product backlog items are needed

Material Required

The list of materials that is required for the releasing planning is as follows:

- Flip charts, markers, whiteboards
- Posted agenda, purpose
- Projector for sharing the data/tools of computers required during a planning meeting
- Planning data

Planning Data

The list of data needed during release planning are as follow:

- Previous iteration data or release planning requests
- Actions plans of previous release/iteration
- Features or defects to be considered
- Organizational and personal calendars
- Velocity from previous releases/ estimates

Output

Following are the output of a release planning:

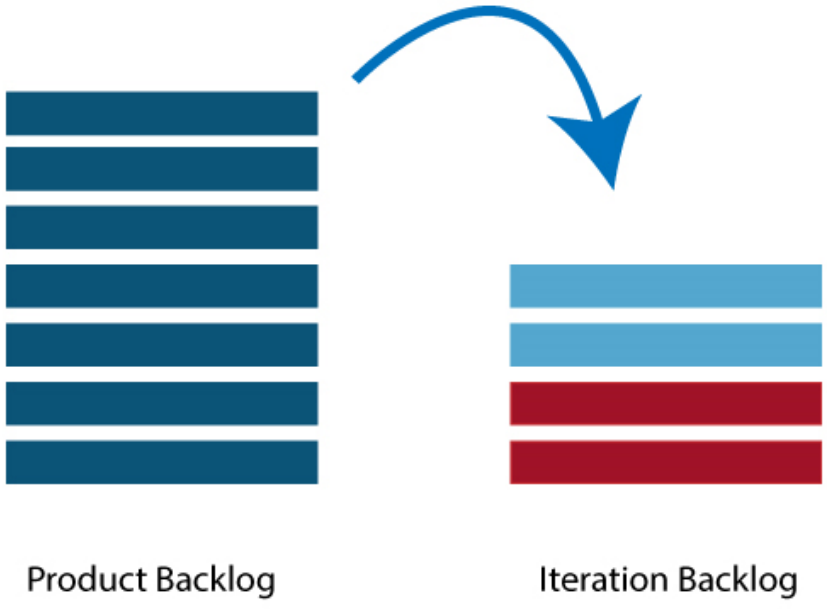
- Release plan
- Commitment
- Issues, dependencies, concerns, and assumptions which are to be monitored
- It suggests improving future release planning

Planning Agenda

- **Opening ceremony:** Welcome message, review purpose, organizing tools, and introduction to business sponsors.
- **Product Vision and Roadmap:** It shows a broad picture of the product.
- **Review previous releases:** Product planning agenda discussed on any item which can impact the plan.
- **Product release name/theme:** It inspects the current status of roadmap themes and makes the necessary adjustment if any.
- **Issues and concerns:** In the agenda, we check any concern or issue, and then record them.
- **Review and Update the Definition of Done:** Review the product build or definition of done and make appropriate changes based on technology.
- **Retrospect:** Require feedback from participants to make the meeting successful.
- **Close:** Celebrate success.

❖ Agile Iteration Planning

The primary purpose of iteration planning is for the team. The team should be a complete set of the top-ranked product backlog items. The completion of top ranked product backlog is a commitment in the time needed on the length of iteration and team velocity.



Who are Involved in the Iteration Planning?

- **Scrum Master:** The Scrum Master is a team leader and facility provider. He helps the team member to follow agile practices so that they can meet their commitments and customers' requirements.
- **Product Owner:** The Product Owner deals with a complete view of the product backlog and their acceptance criteria.
- **Agile Development Team:** Agile delivery defines their tasks and sets the effort. The effort is to estimate the requirements to fulfill the commitment.

Prerequisites of Planning

- The items in the product backlog are sized and have a relative story point assigned.
- The product owner gave the ranking to the portfolio items.
- Acceptance criteria of each portfolio item is clearly stated.

Planning Process

Iteration planning involved the following steps:

- Determines how many requirements (stories) are fit in an iteration.
- Break this requirement into tasks. Assign each task to their owners.
- Each task is set to some estimated time.
- These estimates help the team members to check how many hours for each member will be required to iterate.
- Team members are assigned tasks by seeing their velocity or capacity. Due to this, the team member is not overburdened.

Velocity Calculation

The agile team calculates the velocity based on the previous iterations. A velocity is an average number of units that required finishing user stories in the iteration. Assume that, a team took 10,12,8 story points in each iteration for the previous three iterations, this shows that the team can take 10 as velocity for the next iteration.

Planned velocity tells the team how many user requirements can be completed in the current iteration. If the team instantly finishes the work assigned, then more user requirements can be pulled in. Otherwise, the requirement can be moved out too to the next iteration.

Task Capacity

Three factors determine the capacity of the team:

- Total number of ideal working hours in a day
- A person gives total days in each iteration
- Percentage of time a member is entirely available for the team.

Iteration Planning Steps

- Product Owner describes the highest ranked item of the product backlog.
- Team member describes the tasks required to complete the item.
- Team members own the tasks.
- The team member estimates their own time to finish each task.
- The above steps are repeated for all the items in the iteration.
- If any member is overloaded with work, then his/her tasks distributed among other team members.

❖ Agile Product Backlog

The agile product backlog in Scrum is a list of prioritized features. It contains a short description of all the functionalities desired in the product. In usual scenario, items should be broken down into user stories. Commonly, a Scrum team and its product owner write everything that they can think for agile backlog prioritization.

Why Product Backlog is Important?

- The backlog is prepared to provide an estimate of each feature.
- It helps in the planning of the product's roadmap.
- It helps in the re-ranking the features of the product by adding more value to it.
- It assists in determining the priority of the product first. The team member works first on the higher prioritizes product.

Characteristics of Product Backlog

- Each product should have its own product backlog. It can be a set of large to very large features.
- Multiple team members can work on a single product backlog.
- Ranking of product is based on the technical value, business value, risk management, or strategic fitness.
- Highest priorities items are decomposed into smaller stories during release planning. This is because they can be completed in future iterations.

Product Backlog comprises of the following different types of items

- Features
- Bugs
- Technical work
- Knowledge acquisition

❖ Agile Tools

In agile development, leading as project management is not the easiest job. Jumping between your daily scrums to your next sprint, it causes hard to focus on the work. The agile development tools fulfill your needs, and does it for you.

There are several agile tools available in the market. Some of them are listed below:

- **JIRA Agile**
  Jira is a tool developed by Australian Company Atlassian. It is used for issue tracking, bug tracking, and project management. The bugs and issues are related to your software and Mobile apps. The Jira dashboard consists of many useful functions and features. This function and features make secure handling of issues.

  Features
  - Issue tracking
  - Bug tracking
  - Boards
  - Epics
  - Custom fields

- **ClickUp**
  ClickUp is one of the ultimate agile management tools. It is used for anyone who uses agile methodology. It is the only project management tool whose goal is "to move quickly and easily". CuickUp is in the hand of some of the most famous agile team, including Google and Apple! It is a free forever plan, so, the team can get their hand of ClickUp.

  Features
  - Create epics
  - Use story points
  - Analyze sprint performance
  - Time estimates
  - Start and due dates
  - Time tracking

- **Github**
  Github is one of the largest hosted Git serve where the developers can store all of their codes for a vast number of projects there. The Github provides such a facility of record edits across an entire team in real time. Github is also integrated with many other tools so, many people such as developer and product owner can work on the same code at the same time.
  The project manager can make the Github work for their team. It includes lots of project management tools which help him to inspect what the development team is working on.

  Features
  - Issue tracking
  - Mentions
  - Labels
  - Link issues and pull requests

- **LeanKit**
  LeanKit is the ultimate management tool for a Kanban board on the agile progress for your sprints. It uses cards to represent the work items and live statuses of team member. It works perfect for the remote employees to ensure everyone can see the Kanban board in real time. It prevents the same task to complete twice and make sure the whole team remains on the same page.
  LeanKit work well for cross-functional team which is benefit for Scrum or Kanban boards.
  Features
  - Board view templates
  - Track issues and bugs
  - Manage project portfolios
  - Lean metrics and reporting

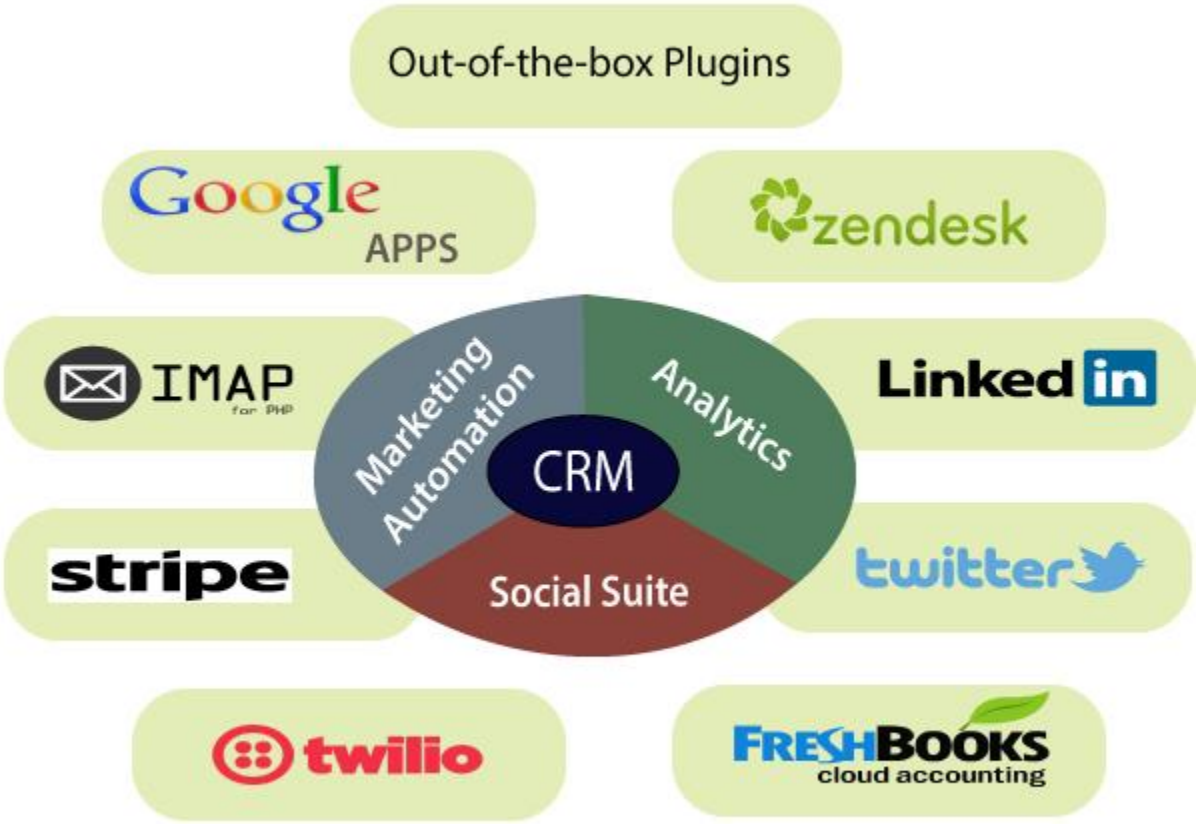
- **Planbox**
  Planbox is a tool that tracks the process of burndown charts. Using this everyone knows how far you are from the Sprint?s completion/goal. Burndown charts are most important part of the agile cycle. Planbox also integrates the customer bug reports, and fixes, making it useful for a wide range of users.
  It has an advance reporting features which make it easy to review the status and areas where improvement is needed at Daily Scrum.

❖ Agile CRM

Agile CRM (Customer Relationship Management) is an All-in-One CRM with Sales, Service automation, and Marketing in a single platform. It consists of sales tracking, marketing automation, contact management, web analytics, telephony, two-way emails, and helpdesk with a simple, clean, and modern interface.

Customer Relationship Management or CRM is software that helps in managing the brand's engagement with your current and future customer.

The CRM software allows you to build, organize, and present database of your customer information. This information can be updated by you and your team when the new data is discovered. It is a central storehouse of all your customer and prospect information which facilitates your team to be organized and more productive. It also allows teamwork among teams and provides management to deeper judgment into individual performance and overall growth of the business.



Why we built Agile CRM?

As an entrepreneur, everyone wants to receive positive feedback and increasing the new product's success. From the first few customers, you engage them with their name, knowing them well enough to talk to them multiple times a day. After that, they may help you to share your product to where it stands. Then, you will get more success, but more trouble also to manage the communication with all of your customers.

Need of Agile CRM

The primary requirement of Agile CRM is broken down into the following points:

1. ***Easy-to-use marketing automation:*** This is easy to use because not every owner of a small businesses should need to understand the technical details of automation.
2. ***A manageable CRM:*** it is easy to use and affordable.
3. ***Telephony:*** There was probably nothing more annoying than seeing someone's phone number on CRM and then dialing their number on our phone. We spent a lot of time in solving our cell phone plans too!
4. ***Social suite:*** Social media playing a vital role in CRM marketing and linking customer with it. If the business owner doesn't incorporate with social media at the very beginning, then they would be left behind.
5. ***Full, two-way email integration:*** The owner wants to integrate personal emails fully into the CRM. So, they can send, receive, and view emails between the owner and their customers inside the CRM itself.

Benefit of Agile CRM

The significant benefit of CRM is that the business moves to the centralized platform to store its data. It makes easy access to information from one common source. Due to the presence of best CRM, the organization gets confident to pay attention to their customer without any additional cost.

Before the availability of CRM, the data has scattered across spreadsheets, documents, address book, notebook, and email system. The CRM, simplify this tedious process, and data are access through a centralized location.