

Agile Practices

Agenda

- History
- What is Agile
- Need of Agile
- How Agile works
- Why we use Agile
- Benefits of Agile



History

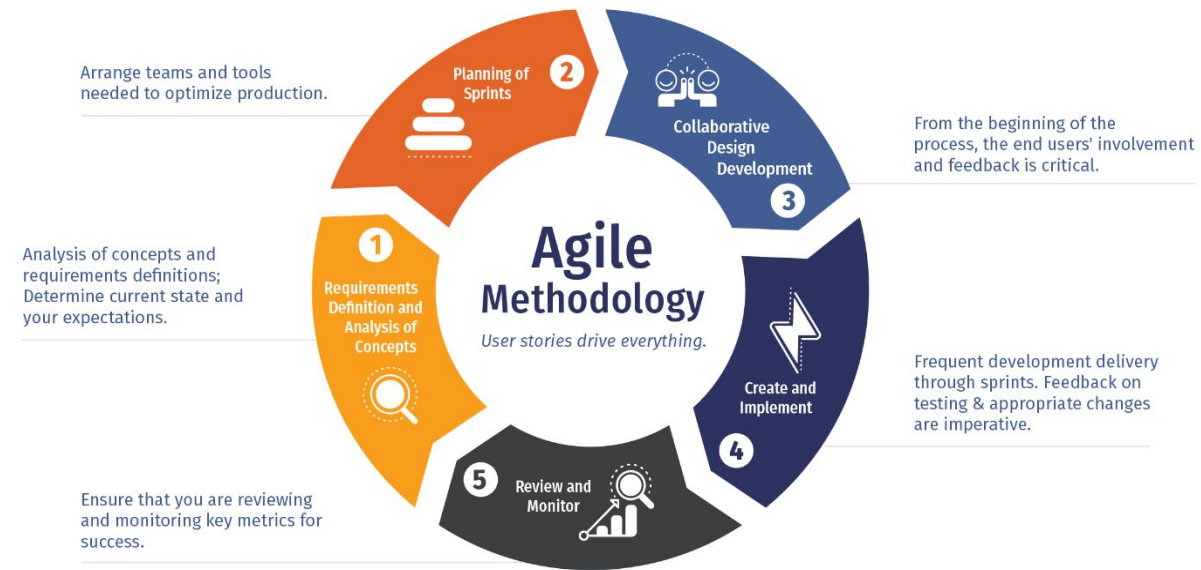
Agile was formally launched in 2001, when 17 technologists drafted the Agile Manifesto. They wrote four major principles for agile project management, intended to guide teams on developing better software:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

What is Agile Methodology?

Agile Methodology meaning a practice that promotes **continuous iteration** of development and testing throughout the software development lifecycle of the project.

In the Agile model in software testing, both development and testing activities are co model.



What is Agile Software Development?

The Agile software development methodology is one of the simplest and effective processes to turn a vision for a business need into software solutions.

Agile is a term used to describe software development approaches that employ continual planning, learning, improvement, team collaboration, evolutionary development, and early delivery. It encourages flexible responses to change.

Need of Agile

Agile is an approach to project management that centers around incremental and iterative steps to completing projects. The incremental parts of a project are carried out in short-term development cycles. **The approach prioritizes quick delivery, adapting to change, and collaboration rather than top-down management and following a set plan.**

In Agile processes, there is constant feedback, allowing team members to adjust to challenges as they arise and stakeholders an opportunity to communicate consistently. Though originally created for software development, the Agile approach is now widely used in executing many different types of projects and in running organizations.

How Agile works

Agile methodology steps

There are 3 main stages of the Agile lifecycle:

1. Concept

The first step of the Agile method is to scope out and prioritize projects. Sit down with your team and stakeholders to brainstorm and identify business opportunities and estimate time and costs to complete each project. Then you can determine which projects are feasible and most valuable and prioritize your project backlog from there.

2. Inception

Once you know what your project is, the next step is to figure out how you will complete it. Who do you need on your team? What are the initial requirements of the customer? Create a diagram to define team responsibilities and scope out the work that needs to be done in each sprint.

3. Iteration

With your initial project defined and approved, the development team can get to work on the first iteration.

The basic workflow during this phase includes:

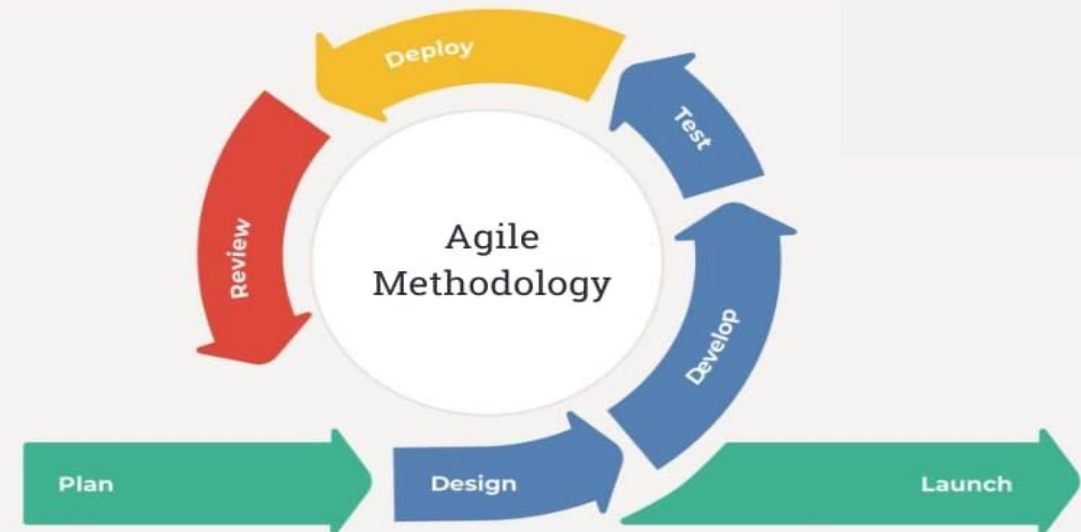
- Requirements—Confirm requirements based on the product backlog and stakeholder feedback.
- Development—Develop the product based on set requirements.
- Testing—Conduct QA testing to validate the features and uncover any issues.
- Delivery—Produce a working product.
- Feedback—Gather feedback from customers and stakeholders in order to define the requirements for the next iteration.

Agile Practices

Why we use Agile Methods

- Improve Customer Involvement
- Increase Quality
- Simply Releases
- Drive Down risk

AGILE METHODOLOGY



Benefits of Agile

1. Increased efficiency

The Agile methodology's success shows us that paperwork and protocol should not be the driving forces behind the creative process. It's quite the opposite — interaction, evaluation, and fluidity foster more motivating and exciting working conditions within cross-functional teams.

2. Satisfied customers

The values and principles of Agile bring the product development cycle back to its roots. It reminds us that products should be designed by people, for people. And it reminds us that we got to where we are now by adapting and improvising based on customer needs.

Agile Practices



Benefits of Agile

3. Improved product quality

Our creative potential flourishes when we have the space to collaborate, think outside the box, iterate, and reshuffle priorities. As a result, you get better products in shorter time frames.

4. Greater flexibility

In an increasingly fast-paced and results-driven world, it no longer makes sense to rigidly define a target. The chances are that by the time you've taken your shot, the target will have moved or become something else entirely.

Waterfall vs. Agile



Agile Practices

Agile Methodologies

- **Scrum** : Scrum is an Agile framework that focuses on cross-functional teamwork, accountability, and iteration in order to develop, deliver, and support complex products. It's primarily used for software development, but its principles can be applied to other project management teams too.

The Scrum framework is organized into key roles, events, and artifacts:

1. Scrum roles

- Product owner
- Scrum master
- Scrum development team



Agile Practices

Agile Methodologies

2. Scrum events

- Daily Scrum
- Sprint planning meeting
- Sprint review
- Sprint retrospective

3. Scrum artifacts

- Product backlog
- Sprint backlog
- Increment (or Sprint Goal)

Scrum teams use tools like scrum task boards to help organize tasks and sprints to help team members visualize the current status of projects.

Agile Practices

Agile Methodologies

- **Extreme Programming** : Extreme Programming (XP) is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development.
- **Dynamics System Development Method** : DSDM is an Agile method that focuses on the full project lifecycle, DSDM (formally known as Dynamic System Development Method) was created in 1994, after project managers using RAD (Rapid Application Development) sought more governance and discipline to this new iterative way of working.

Agile Practices

Agile Methodologies

The Eight Principles of DSDM:

1. Focus on the business need
2. Deliver on time
3. Collaborate
4. Never compromise quality
5. Build incrementally from firm foundations
6. Develop iteratively
7. Communicate continuously and clearly
8. Demonstrate control

Agile Methodologies

Feature Driven Development : An Agile methodology for developing software, Feature-Driven Development (FDD) is customer-centric, iterative, and incremental, with the goal of delivering tangible software results often and efficiently. FDD in Agile encourages status reporting at all levels, which helps to track progress and results.

FDD allows teams to update the project regularly and identify errors quickly. Plus, clients can be provided with information and substantial results at any time. FDD is a favorite method among development teams because it helps reduce two known morale-killers in the development world: Confusion and rework.

Agile Practices

Agile Methodologies

Kanban : Kanban is an agile model designed to help teams work together more effectively. It follows three guiding principles:

- Visualize your workflow.
- Limit the amount of work in progress.
- Organize the workflow based on priority.

Unlike Scrum, Kanban doesn't have prescribed roles or timeboxed sprints.

Instead, Kanban focuses on shorter cycles for faster delivery and transparency throughout development so everyone understands who is responsible for what and when.



Introduction to JIRA

Agenda

- What is JIRA
- How to use JIRA
- JIRA Concepts
- Features
- Why JIRA

**What is
Jira software
and what is it
used for?**



Introduction to JIRA

What is JIRA

JIRA is a tool developed by Australian Company Atlassian. This software is used for **bug tracking, issue tracking, and project management**.

The JIRA full form is inherited from the Japanese word “Gojira” which means “Godzilla”. The basic use of this tool is to track issue and bugs related to your software and Mobile apps.

It is also used for project management. The JIRA dashboard consists of many useful functions and features which make handling of issues easy.

Introduction to JIRA

How to Use JIRA?

Here is a step-by-step process on how to use Jira software:

Step 1) Open Jira software and navigate to the Jira Home icon

Step 2) Select Create project option

Step 3) Choose a template from the library

Step 4) Set up the columns as per your need from Board settings

Step 5) Create an issue

Step 6) Invite your Team members and start working

Introduction to JIRA

Why is Jira so popular?

There are many reasons why Jira is popular and considered amongst the top project management tools available today.

- 1. Jira is designed for agile team :** Jira is based on Agile project management, which allows companies like Microsoft and Google to update their product very often, as opposed to taking years!
- 2. Jira is good at tracking projects :** Jira is known as an issue tracking software that can track ongoing projects at any stage. It allows you to keep track of all activities, including issues and their updates, people assigned, team comments, and so on, which allows for greater visibility and collaboration across teams and departments.

3. You can track time in Jira with colors :

Jira uses 3 colors (blue, orange and green) to track the amount of time spent on any given issue; you get access to this information in the time tracking section.

- **Blue** shows the time estimated to complete the issue
- **Orange** denotes the amount of time left for fixing the issue
- **Green** represents the actual time spent on resolving the issue

4. Reports in Jira are great

There are many reports available in Jira, to offer you project statistics throughout the entire lifecycle; pie chart report, user workload report, cumulative flow diagram report, sprint report, burndown chart report are all examples of reports you can create in Jira.

5. Jira offers many search options

With Jira, you can find what you're looking for quite easily; you can, for example, save specific filters to use again. Some of the searching functionalities you'll find in Jira are simple search, quick search, export search, configurable search results, advanced search, search status history, and refine searches.

6. Jira security ensures work is clear

Security levels can be assigned to various team members so only they are allowed to work on certain tasks, like bug tracking for example. Also, the permission schemes allow you to make a set of permissions for any project; components for example are usually set in such a way that only project admins can change them

Introduction to JIRA

JIRA Features

- **Usable** : Intuitive interface designed with both business and technical users in mind.
- **Track** : Keep tracks of all activities, changes and work logged against issues.
- **Administration** : A low maintenance system with straightforward administration activities
- **Extensible** : With over 100 plugins contributed by the community
- **Open** : An open API, full source code access allow for further integration and customization of JIRA functionality.

Introduction to JIRA

JIRA Concepts

1. Project

- A JIRA Project is a collection of issues.
 - Every issue belongs to a project .
- Each Project is identified by a name and Key.
- Project key is appended to each issue associated with the project

Example :

- Name of project : Social Media
 - Project key : SM
- Issue : SM_24 Add n new friend

Introduction to JIRA

JIRA Concepts

2. What is an Issue ?

Issue is the building block of the project

Depending on how your organization is using JIRA, an issue could represent :

- A software bug
- A project task
- A helpdesk ticket
- A product improvement
- A leave Request from client.

Introduction to JIRA

JIRA Concepts

3. What is a Component ?

Components are a sub section of a project.

Components are used to group issues within a project to smaller parts.

4. What is a Workflow ?

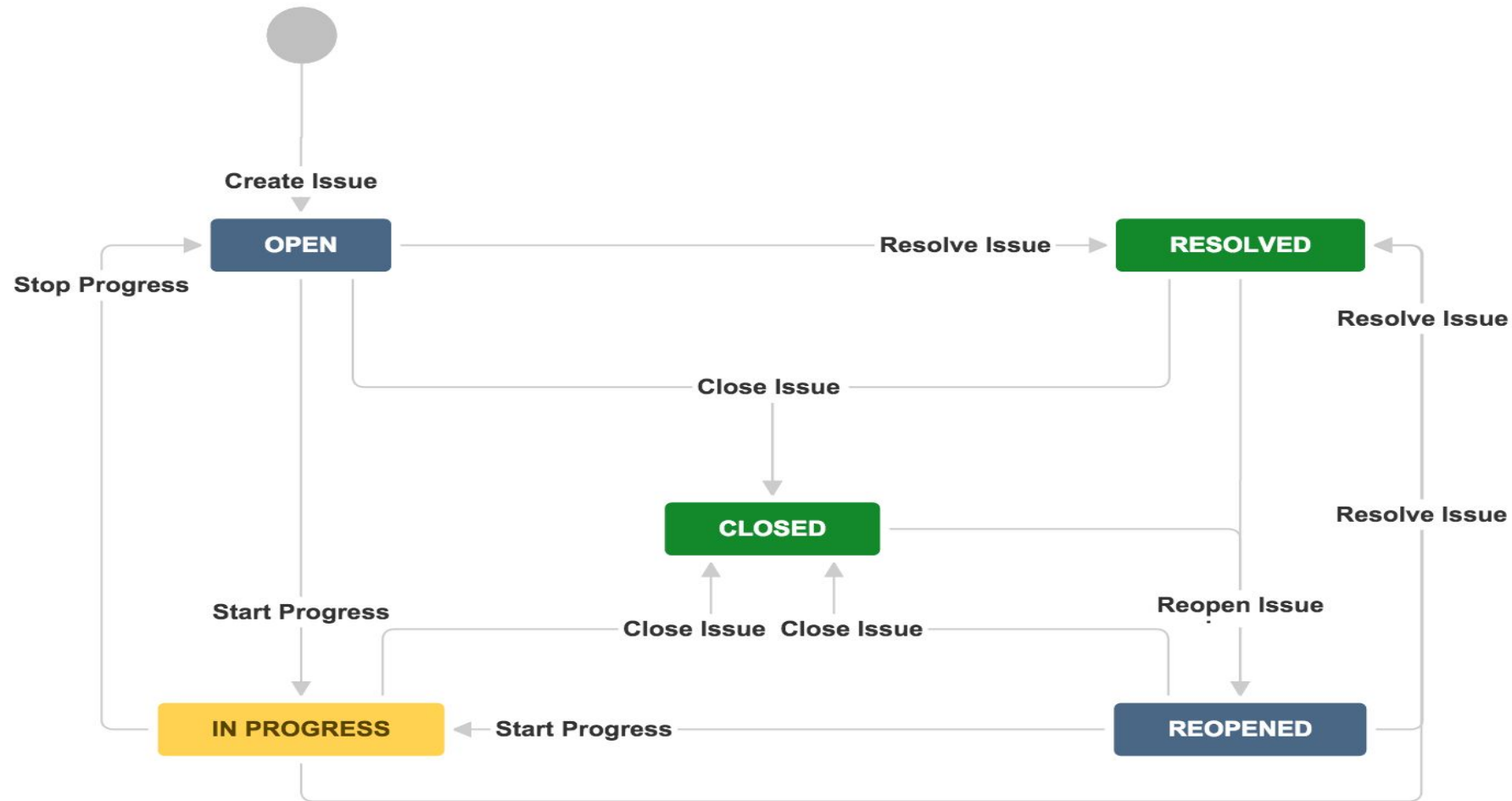
A JIRA Workflow is the set of statuses and transitions that an issue goes through during its lifecycle.

Workflow typically represents business processes.

JIRA comes with a default workflow, and it can be customized to fit your organization

Introduction to JIRA

JIRA'S Default Workflow



Introduction to JIRA

What is Jira used for?

Jira Software is part of a family of products designed to help teams of all types manage work. Originally, Jira was designed as a bug and issue tracker. But today, Jira has evolved into a powerful work management tool for all kinds of use cases, from requirements and test case management to agile software development.

Jira for requirements & test case management

An increasing number of teams today are developing more iteratively, and Jira Software is the central hub for the coding, collaboration, and release stages. For test management, Jira integrates with a variety of add-ons, so the QA's testing slides seamlessly into the software development cycle. Teams can test effectively and iteratively. QA teams use Jira issues, customized screens, fields, and workflows to manage manual and automated tests.