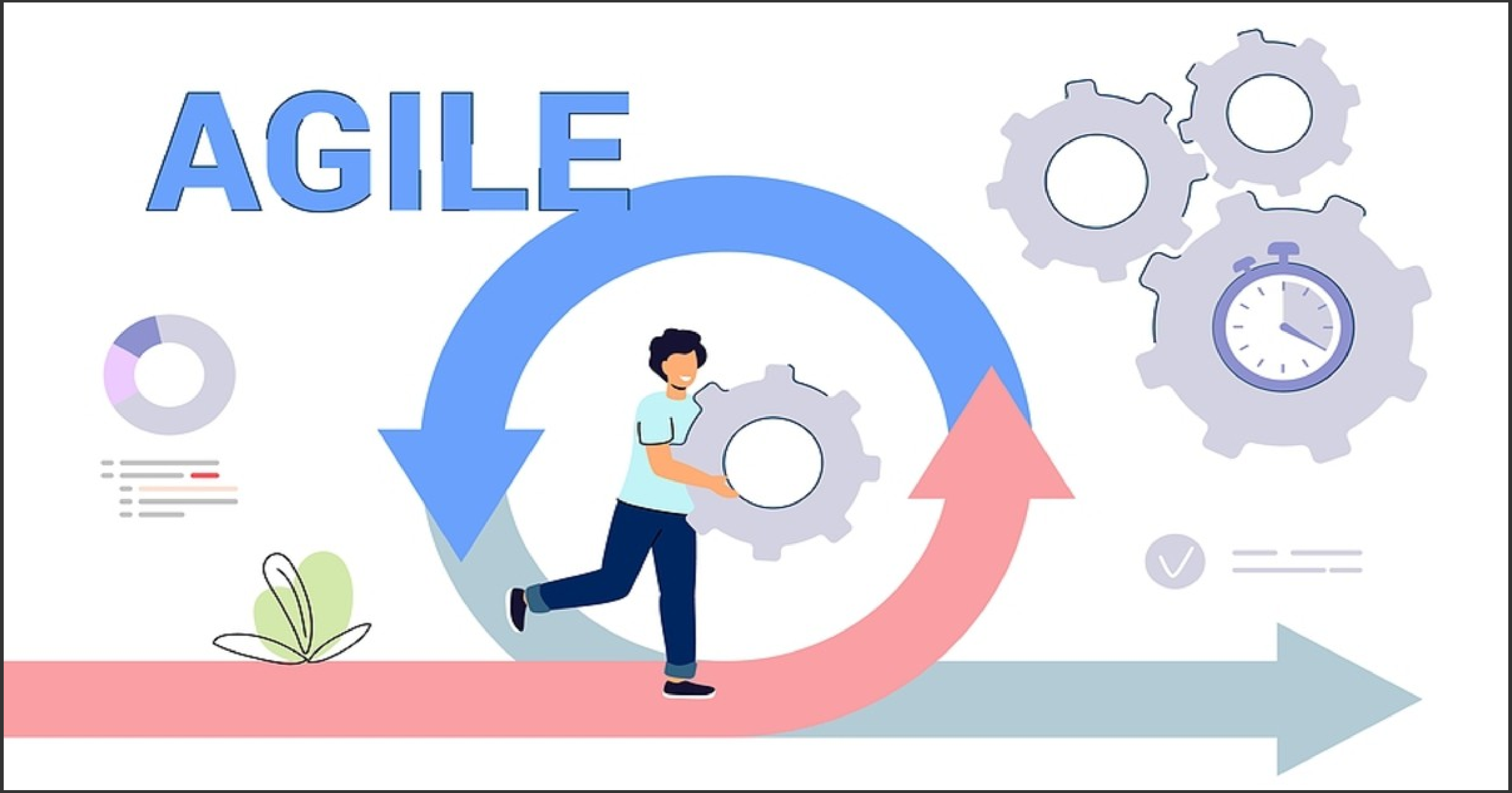
Agile SDLC



Development Phases

- Planning / Design

Defining the goals and requirements for the project, however agile leaves much of this phase open-ended, allowing for flexibility in changing conditions of the project.

- Develop

The features are coded and implemented within the project.

- Testing

The implemented features are examined for bugs and inconsistencies.

- Deployment/Launch

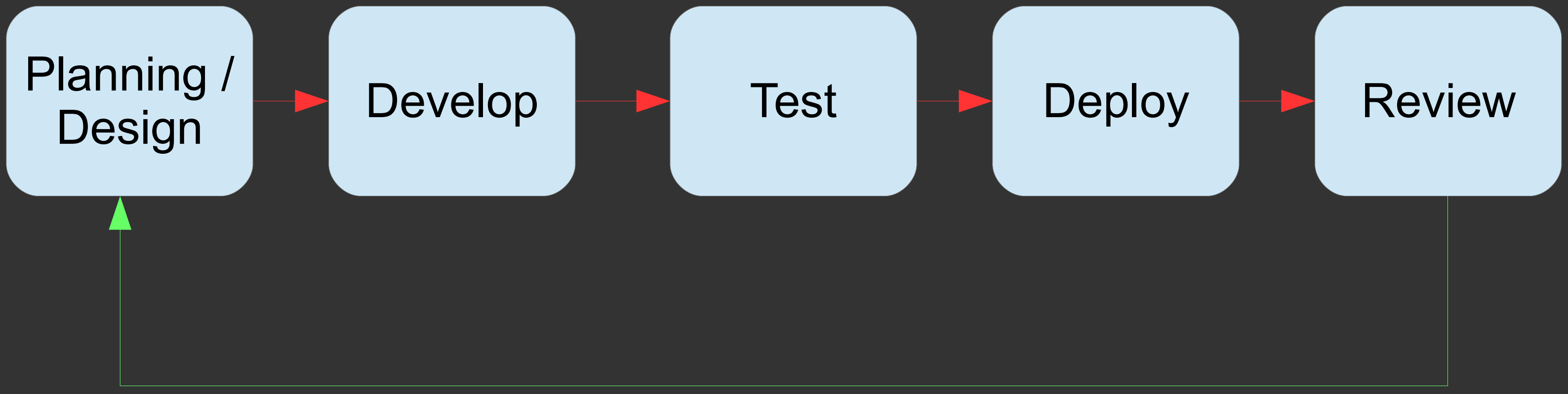
The current version of the project is released / made available.

- Review

The teams will review both the process of the iteration's development, as well as the feedback from stakeholders and users (in the form of user stories) to better inform their approach to the next iteration.

Iterations

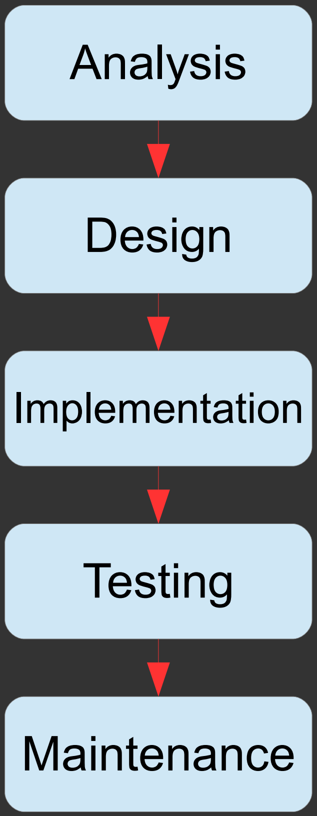
Iterations are a fundamental part of agile, they are standardized, relatively short time-frames in which the product is cycled through each phase of the development process (Iterations, 2022) . A project will go through multiple iterations before its final completion, the idea being that a continuous cycle of iterations delivers a usable product faster and allows for adjustments to be made depending on how the last version is received by users and stakeholders. The diagram below is a basic display of an iteration.



Cycle repeats for next iteration

Contrasting Agile and Waterfall Development Cycles

In the context of agile, the waterfall method could be considered as a project completed with a single iteration. It is a linear process that follows the diagram below.



Analysis

Design

Implementation

Testing

Maintenance

- Each step in the waterfall method is distinct and separate.
- A critical difference between the two approaches is that waterfall relies upon *extensively detailed* requirements before any development begins.
  (Cobb, 2015, p.4)
- “All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name 'Waterfall Model'".

(TutorialsPoint, 1)

Agile, or Waterfall?

So how might you discern which approach is best suited to your project? Considering the following questions should help to make this decision – are you able to near-completely delineate the functionality and features of your project? Or will a majority of the features be subject to change? How much emphasis will be placed upon user feedback? Are the stakeholders willing to wait until development is nearly complete until they are presented with the product?

Consider Agile

- Features are subject to change, or not completely and concretely defined.
- Feedback from users and stakeholders will dictate the direction of the product.
- Stakeholders prefer to be frequently updated and to evaluate the project as it is developed.

Consider Waterfall

- Features are able to be near-completely defined at the outset.
- User feedback will not be a primary consideration in how the project will fulfill the project's goals set by the stakeholders.
- A cohesive and extensive plan to follow makes the linear development process and its straight-forwardness desirable.

Scrum-Agile Team Roles

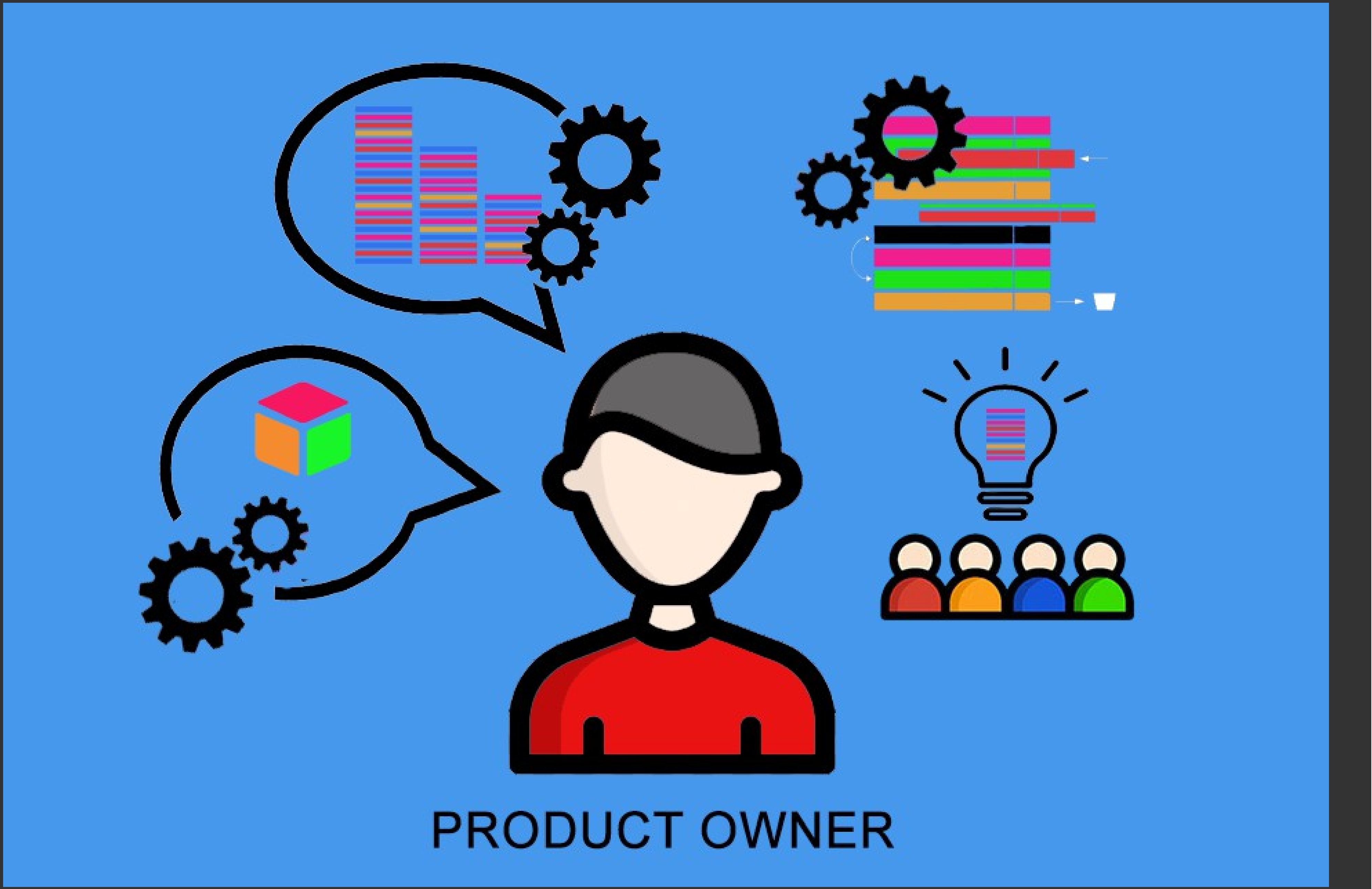
Product Owner

Scrum Master

Developer

Section: Agile Team Roles

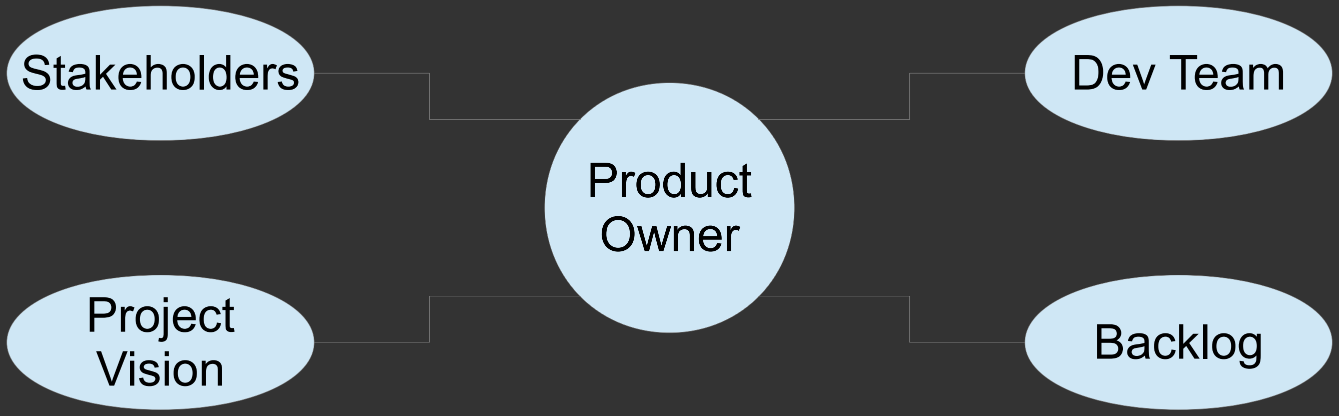
Product Owner



Section: Agile Team Roles
Role: Product Owner

Overview

The product owner is a role within the agile framework that is tasked with organizing and delineating the work undertaken by the broader team. They serve as an intermediary and vital line of communication between the stakeholders and the development team working on the implementation of the product's vision (Overeem, 2). They function as a sort of arbiter and locus for the project, represented by the diagram below:



Section: Agile Team Roles
Role: Product Owner

Responsibilities

- Clearly define and communicate the product's vision.

  By explicitly framing the project's goals and functionality, the project owner allows the team to maintain a clear picture of the project's criteria for success. A well communicated product vision will keep every member of the team on the same page.

- Prioritize and maintain the project's backlog

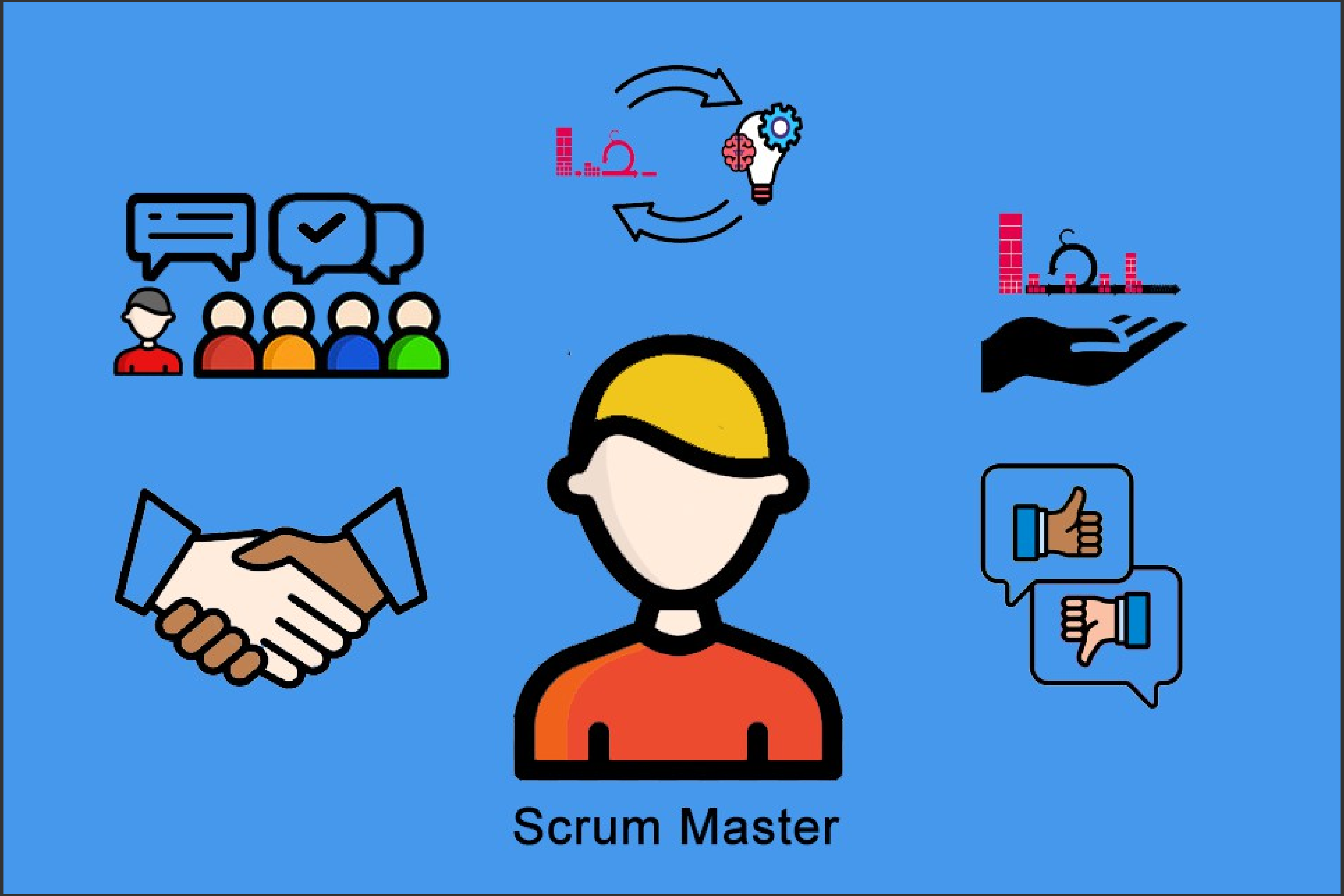
  Assorting the project's backlog in terms of each item's “priority, risk, and value” (Overeem, 2) allows the development process to focus on vital components of the project, maximizing value during each iteration of development. Effective Management of the backlog by the project manager acts as a guiding force toward the development team's efforts.

- Act as a liaison between stakeholders and development team

  By integrating the feedback received from stakeholders and users (in the form of user stories), the project manager will make the necessary changes and adjustments to the backlog and vision according to the shifting demands of both market-driven and client-based factors.

Section: Agile Team Roles

Scrum Master



Section: Agile Team Roles
Role: Scrum Master

Overview

The scrum master, similarly to the product owner, is there to assist the development team in their efforts to produce tangible progress in each iteration of the project. Although their prerogative differs in a very significant way, in that they are not there to dictate or provide project-specific instructions – rather they act as a mentor in teaching and guiding the team through the workflow inherent to the agile framework. A scrum master will have a strong working relationship with the product owner, and will effectively integrate the directives they receive into the scrum events that they are tasked with coordinating. Through adherence to the principles of the scrum-agile framework, and effectively guiding team members through the process – scrum masters create an environment where the project stands to gain the most from the unique approach to development that agile offers. The importance of a scrum master is pronounced in situations where a team is new/unfamiliar to the agile framework, where their efforts as a guide/mentor are particularly useful in creating a working environment suitable for the agile framework (Schwaber & Sutherland, 6).

Section: Agile Team Roles
Role: Scrum Master

Responsibilities

- Acts as a resource and mentor/guide for effective implementation of agile development.

  Scrum masters must assist each team member in their efforts to work within the agile framework. Whether it's working with the product owner to better understand the best techniques for the management of the backlog, or guiding the daily scrum meetings to ensure that the team's time is spent wisely – the scrum master will have to maintain a certain degree of control over the working culture in order to fully take advantages of agile.

- Coordinates and facilitates scrum events.

  Whether it be a daily scrum, or a sprint planning / retrospective session, the scrum master will be vital to the pacing and productiveness of the event. Through effective communication and mentorship, they allow the team to work effectively.

- Understand and relay the importance of scrum/agile's core principles.

  Scrum masters will instill the principles of collaboration, interoperability, within the team, and foster a culture which embodies the values of commitment, openness, respect, and a sense of collective ownership over the project. (Cobb, 36)

Section: Agile Team Roles

Developer

```
Category.delete_all; [illegible]
Shoulda::Matchers.configure do |config|
  config.integrate do |with|
    with.test_framework :rspec
    with.library :rails
  end
end

# Add additional requires below this line. [illegible]

# Requires supporting ruby files [illegible]
# spec/support/ and its [illegible]
# run as spec files by default. [illegible]
# in _spec.rb will both be [illegible]
# run twice. It is [illegible]
# end with _spec.rb. You can [illegible]
```

Section: Agile Team Roles
Role: Developer

Overview

The development team is ultimately who will be working on the nuts-and-bolts functionality and implementation of the project. Although some members will have certain areas of expertise (testers, and designers for instance), within the agile framework “everyone is a developer, no one has a special title.”(Overeem, 9) this de-structuring is meant to put a special emphasis on agile's commitment to the values of collaboration and interoperability, as well as fostering a sense of collective ownership over the project. Within agile, development teams are allowed a good deal of latitude to direct their approach towards completing the project's goals, they are meant to be a self-organizing group capable of understanding the challenges being faced, and be able to implement solutions with minimal oversight/micro-management or explicit instruction.

Section: Agile Team Roles
Role: Developer

Responsibilities

- Creates and delivers features during each sprint/iteration.

  The main driver of agile's success is its ability to churn out functionality and improvements in a relatively short time-frame. Although much of the responsibility in terms of addressing which specific features and items should be addressed in any particular iteration, it ultimately falls upon the development team to get it done, and ideally create a “flow of continuous delivery [of features].” (Overeem, 10)

- Effectively addresses items within the project's backlog

  Although the development team has an enhanced degree of autonomy, they must always bear in mind the backlog and the priority of certain items within it as designated by the product owner.

- Adheres to agile's development procedures/guidelines

  Effective self-organization of the development team depends upon their ability to communicate, operate as a single unit – where no particular member is siloed, and maintains and open and honest tenor of discussions among themselves.

References

Iterations, (2022, December).
Scaledagileframework.
https://scaledagileframework.com/iterations/

Cobb, Charles G. (2015).
The Project Manager's Guide to Mastering Agile : Principles and Practices for an Adaptive Approach. Wiley.

SDLC – Waterfall Model,
Tutorials point,
http://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm

Overeem, Barry (2016).
Characteristics of a Great Scrum Team, Scrum.org
https://scrumorg-website-prod.s3.amazonaws.com/drupal/2016-08/Characteristics%20of%20a%20Great%20Scrum%20Team.pdf

Schwaber, Ken. Sutherland, Jeff. (2020, November)
The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game