

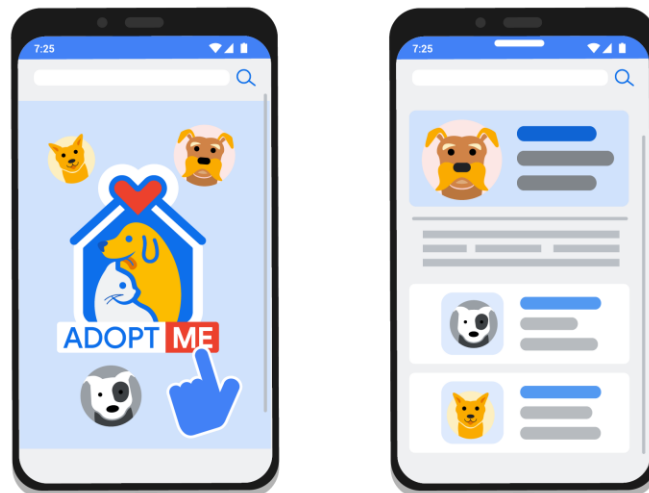
Releasable Increment versus Minimum Viable Product

You have learned that the **Product Increment** is what is produced after a given Sprint and is considered releasable. The Scrum Guide states that “An Increment is a concrete stepping stone toward the Product Goal. Each Increment is additive to all prior Increments and thoroughly verified, ensuring that all Increments work together. In order to provide value, the Increment must be usable.”

Potentially releasable product increment

Potentially **releasable** (or shippable) Product Increment is a handy way for teams to think about the desired result of a Sprint. The goal for every Sprint is to result in a completed, tested, ready-to-ship addition to the product or solution. This doesn’t mean the product will actually ship to customers—that is why they use the word “potentially.” Consider, for example, building an application for finding and adopting pets. Three features on the product backlog could be:

1. Presenting information about available pets
2. Rating potential matches based on adopters
3. Allowing user to contact the adoption center



It doesn’t make sense to ship any one of those features in isolation, but finishing a potentially releasable Product Increment is helpful because it enables the team to see one

feature in its entirety rather than to work on bits and pieces of all three features with none of them actually completed. With a potentially releasable Product Increment, you will create a complete, working, tested implementation of one feature in a single sprint.

Having a potentially releasable Increment also allows the team to get early feedback on the product, ensure that the work is high quality, and have the opportunity to respond to change. You should always focus on a potentially releasable product increment as your sprint goal.

Definition of Done

A related term is the **Definition of Done**. The Definition of Done is a formal description of the state of the potentially releasable Product Increment and what it means when it meets the quality measures required for the product. It is the team's agreed-upon requirements for any backlog item that is considered "done." In software projects, teams often decide that "done" means the software has been completed, reviewed, and has passed testing. In a non-software project, a Definition of Done may be a document including a legal review with approval or a formalized closeout report. The important part of figuring out your team's Definition of Done is to have an explicit, shared understanding of what being "done" entails.

But how do you know when a solution is shippable or releasable? In a Scrum Team, it is ultimately the decision of the Product Owner to ensure there is value before releasing an item. To determine this, they may consider a few things:

1. Is the increment complete?
2. Will it bring value and does it meet quality measures? Has it been well-tested?
3. Is it usable by the end user? Can we use their direct or indirect feedback to improve future versions of the product?

Comparing Releasable Product Increment to Minimum Viable Product

As you learned in the previous video, a **minimum viable product (MVP)** is a version of a product with just enough features to satisfy early customers. Eric Ries, an entrepreneur and author, coined the term in this [guide](#) and defined an MVP as "that version of a new product which allows a team to collect the maximum amount of validated learning about customers with the least effort." In other words, gathering insights from an MVP enables quicker feedback from users than developing a full-featured product that may not be 100% tested or secure. Some examples of an MVP could be a landing page for your website or a "buy now" button that doesn't do anything other than register that someone has clicked it. A minimum viable product is a package of features that may take several sprints to

develop, but every sprint's goal is to produce a **product increment**. To differentiate between a potentially releasable increment and a MVP, let's take our example of the online pet adoption app and the three features we discussed previously. We noted that each of these features on their own wasn't a useful release of the solution. However, the Product Owner may decide that the MVP for this user experience is to implement these three requirements for cats only. By reducing the scope of the MVP, the Product Owner is able to release the solution into the marketplace and collect feedback from the users who wish to adopt cats. This feedback will be valuable not only for the cat adoption process but for any type of pet adoption in future iterations of the product.

Key takeaway

So can a releasable increment be an MVP? Yes! Does it always have to be an MVP? Not necessarily. A Scrum Master or Product Owner is always making sure that the team is building potentially releasable increments of the solution or product. Then, the Product Owner uses those product increments and business insights to determine what will make up a valuable and viable release of the product to their customers. This is based on both user value delivered and the ability to gather feedback that will continuously improve the product.