



Product Enterprise Resources Blog Support
Sign In Try It Free

MoSCoW Prioritization

What is MoSCoW Prioritization?

MoSCoW prioritization, also known as the MoSCoW method or MoSCoW analysis, is a popular prioritization technique for managing requirements.

The acronym MoSCoW represents four categories of initiatives: must-have, should-have, could-have, and won't-have, or will not have right now. Some companies also use the “W” in MoSCoW to mean “wish.”

What is MoSCoW Prioritization Method? Definition, Overview, Examples, & More



What is the History of the MoSCoW Method?

Software development expert Dai Clegg created the MoSCoW method while working at Oracle. He designed the framework to help his team prioritize tasks during development work on product releases.

You can find a detailed account of using MoSCoW prioritization in the [Dynamic System Development Method \(DSDM\) handbook](#). But because MoSCoW can prioritize tasks within any

time-boxed project, teams have adapted the method for a broad range of uses.

How Does MoSCoW Prioritization Work?

Before running a MoSCoW analysis, a few things need to happen. First, key stakeholders and the product team need to get aligned on objectives and prioritization factors. Then, all participants must agree on which initiatives to prioritize.

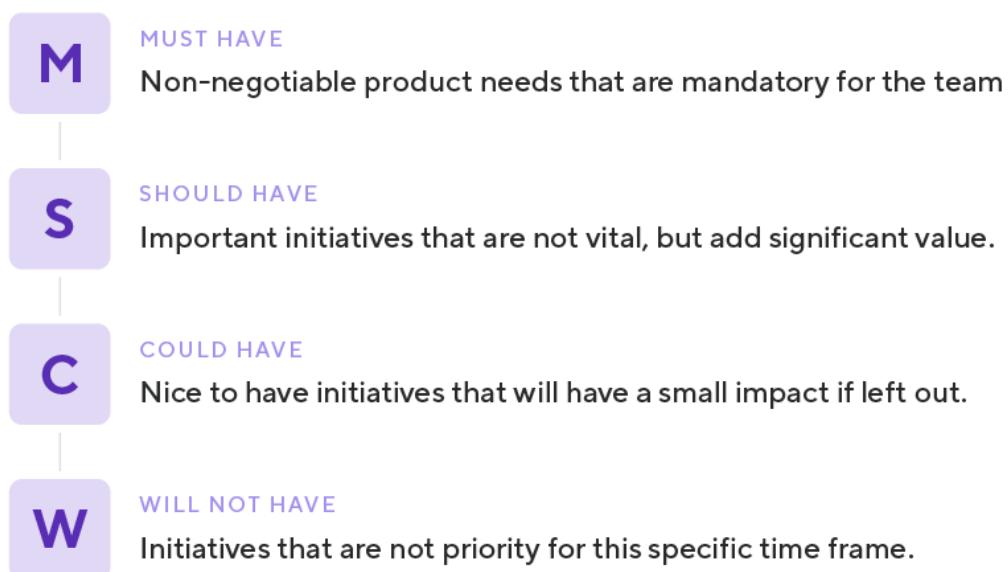
At this point, your team should also discuss how they will settle any disagreements in prioritization. If you can establish how to resolve disputes before they come up, you can help prevent those disagreements from holding up progress.

Finally, you'll also want to reach a consensus on what percentage of resources you'd like to allocate to each category.

With the groundwork complete, you may begin determining which category is most appropriate for each initiative. But, first, let's further break down each category in the MoSCoW method.

[Download Prioritize Your Roadmap →](#)

MoSCoW Prioritization Categories



1. Must-have initiatives

As the name suggests, this category consists of initiatives that are “musts” for your team. They represent non-negotiable needs for the project, product, or release in question. For example, if you’re releasing a healthcare application, a must-have initiative may be **security functionalities** that help maintain compliance.

The “must-have” category requires the team to complete a mandatory task. If you’re unsure about whether something belongs in this category, ask yourself the following.

Must Have Initiatives

- What will happen if this initiative is **not included** in this specific release?
- Is there a **simpler** way to accomplish this?
- Will the product work **without** it?

 ProductPlan

If the product won’t work without an initiative, or the release becomes useless without it, the initiative is most likely a “must-have.”

2. Should-have initiatives

Should-have initiatives are just a step below must-haves. They are essential to the product, project, or release, but they are not vital. If left out, the product or project still functions. However, the initiatives may add significant value.

“Should-have” initiatives are different from “must-have” initiatives in that they can get scheduled for a future release without impacting the current one. For example, performance improvements, minor bug fixes, or new functionality may be “should-have” initiatives. Without them, the product still works.

3. Could-have initiatives

Another way of describing “could-have” initiatives is nice-to-haves. “Could-have” initiatives are not necessary to the core function of the product. However, compared with “should-have” initiatives, they have a much smaller impact on the outcome if left out.

So, initiatives placed in the “could-have” category are often the first to be deprioritized if a project in the “should-have” or “must-have” category ends up larger than expected.

4. Will not have (this time)

One benefit of the MoSCoW method is that it places several initiatives in the “will-not-have” category. The category can manage expectations about what the team will not include in a specific release (or another timeframe you’re prioritizing).

Placing initiatives in the “will-not-have” category is one way to help prevent [scope creep](#). If initiatives are in this category, the team knows they are not a priority for this specific time frame.

Some initiatives in the “will-not-have” group will be prioritized in the future, while others are not likely to happen. Some teams decide to differentiate between those by creating a subcategory within this group.

How Can Development Teams Use MoSCoW?

Although Dai Clegg developed the approach to help prioritize tasks around his team’s limited time, the MoSCoW method also works when a development team faces limitations other than time. For example:

Prioritize based on budgetary constraints.

What if a development team’s limiting factor is not a deadline but a tight budget imposed by the company? Working with the product managers, the team can use MoSCoW first to decide on the initiatives that represent must-haves and the should-haves. Then, using the development department’s budget as the guide, the team can figure out which items they can complete.

Prioritize based on the team’s skillsets.

A cross-functional product team might also find itself constrained by the experience and expertise of its developers. If the product roadmap calls for functionality the team does not have the skills to build, this limiting factor will play into scoring those items in their MoSCoW analysis.

Prioritize based on competing needs at the company.

Cross-functional teams can also find themselves constrained by other company priorities. The team wants to make progress on a new product release, but the executive staff has created tight deadlines for further releases in the same timeframe. In this case, the team can use MoSCoW to determine which aspects of their desired release represent must-haves and temporarily backlog everything else.

What Are the Drawbacks of MoSCoW Prioritization?

Although many product and development teams have prioritized MoSCoW, the approach has potential pitfalls. Here are a few examples.

1. An inconsistent scoring process can lead to tasks placed in the wrong categories.

One common criticism against MoSCoW is that it does not include an objective methodology for ranking initiatives against each other. Your team will need to bring this methodology to your analysis. The MoSCoW approach works only to ensure that your team applies a consistent scoring system for all initiatives.

Pro tip: One proven method is weighted scoring, where your team measures each initiative on your backlog against a standard set of cost and benefit criteria. [You can use the weighted scoring approach in ProductPlan's roadmap app.](#)

2. Not including all relevant stakeholders can lead to items placed in the wrong categories.

To know which of your team's initiatives represent must-haves for your product and which are merely should-haves, you will need as much context as possible.

For example, you might need someone from your sales team to let you know how important (or unimportant) prospective buyers view a proposed new feature.

One pitfall of the MoSCoW method is that you could make poor decisions about where to slot each initiative unless your team receives input from all relevant stakeholders.

3. Team bias for (or against) initiatives can undermine MoSCoW's effectiveness.

Because MoSCoW does not include an objective scoring method, your team members can fall victim to their own opinions about certain initiatives.

One risk of using MoSCoW prioritization is that a team can mistakenly think MoSCoW itself represents an objective way of measuring the items on their list. They discuss an initiative, agree that it is a “should have,” and move on to the next.

But your team will also need an objective and consistent framework for ranking all initiatives. That is the only way to minimize your team’s biases in favor of items or against them.

When Do You Use the MoSCoW Method for Prioritization?

MoSCoW prioritization is effective for teams that want to include representatives from the whole organization in their process. You can capture a broader perspective by involving participants from various functional departments.

Another reason you may want to use MoSCoW prioritization is it allows your team to determine how much effort goes into each category. Therefore, you can ensure you’re delivering a good variety of initiatives in each release.

What Are Best Practices for Using MoSCoW Prioritization?

If you’re considering giving MoSCoW prioritization a try, here are a few steps to keep in mind. Incorporating these into your process will help your team gain more value from the MoSCoW method.

1. Choose an objective ranking or scoring system.

Remember, MoSCoW helps your team group items into the appropriate buckets—from must-have items down to your longer-term wish list. But MoSCoW itself doesn’t help you determine which item belongs in which category.

You will need a separate ranking methodology. You can choose from many, such as:

- Weighted scoring
- Value vs. complexity
- Kano model
- Buy-a-feature
- Opportunity scoring

For help finding the best scoring methodology for your team, check out ProductPlan's article: [7 strategies to choose the best features for your product.](#)

2. Seek input from all key stakeholders.

To make sure you're placing each initiative into the right bucket—must-have, should-have, could-have, or won't-have—your team needs context.

At the beginning of your MoSCoW method, your team should consider which stakeholders can provide valuable context and insights. Sales? Customer success? The executive staff? Product managers in another area of your business? Include them in your initiative scoring process if you think they can help you see opportunities or threats your team might miss.

3. Share your MoSCoW process across your organization.

MoSCoW gives your team a tangible way to show your organization prioritizing initiatives for your products or projects.

The method can help you build company-wide consensus for your work, or at least help you show stakeholders why you made the decisions you did.

Communicating your team's prioritization strategy also helps you set expectations across the business. When they see your methodology for choosing one initiative over another, stakeholders in other departments will understand that your team has thought through and weighed all decisions you've made.

If any stakeholders have an issue with one of your decisions, they will understand that they can't simply complain—they'll need to present you with evidence to alter your course of action.

Related Terms

[2x2 prioritization matrix](#) / [Eisenhower matrix](#) / [DACI decision-making framework](#) / [ICE scoring model](#) / [RICE scoring model](#)

[Download Prioritizing Your Product Roadmap →](#)

Try ProductPlan Free for 14 Days

Get started in minutes—no credit card required.

[Try It Free](#)

[Product](#)

[Features](#)

[Pricing](#)

[Support](#)

[Enterprise](#)

[Integrations](#)

[Templates](#)

[Security](#)

[Release Notes](#)

[Status](#)

[Company](#)

[About Us](#)

[Contact Us](#)

[Careers](#)

[Privacy Policy](#)

[Terms of Service](#)



Resources

[Learning Center](#)

[Blog](#)

[Glossary](#)

[Downloads](#)

[Webinars](#)

Ultimate Guides

[Product Roadmaps](#)

[Product Managers](#)

[Product Management](#)

[Product Planning](#)

[Product Management Frameworks](#)

[Product Strategy](#)

[Resources for Product Managers](#)

[Product Launch](#)



© ProductPlan 2022