APPFOLLOW

REVIEW MANAGEMENT HANDBOOK 2025

Basics | Tactics | Automation

Please avoid sharing the content of this guide without crediting AppFollow.
© AppFollow.fi Oy (Finland), 2025

Table of contents

Introduction 4

How does review management work? 5

Comparison to app store consoles 9

Review types — which need more attention? 13

Metrics — what to track? 21

AppFollow best practices 26

"Only about 10% of your users will ever leave a review, but that vocal minority determines whether the silent 90% download your app or skip it entirely. Cross below 4.0 stars and you're suddenly fighting for table scraps in the app store economy."



Anatoly Sharifulin
CEO & co-founder, AppFollow

Introduction

Review management is simple at the surface level. You see a review, you respond. If there are many, you categorize, look for patterns, and learn what your users are telling you at scale. There's no need to go through every single review, but it sure helps to find some gems that'll require human intervention.

How about sentiment? Say you're choosing between a 4.1 and 4.2-star app. You check out what other people are saying about the app to gauge if you like it or not.

How about this: **would you choose a 3.9 app or a 4.4 app?** Whichever drives more trust?

From a user's standpoint, these are all more or less simple decisions. One would be pleasantly surprised to see developers respond in the reviews. A dev team that cares, in 2025? Unheard of.

From an app's view, things are complicated, of course they are. Different feedback requires different approaches. Some are more automatable than others. At the core of it all, a very human thing to do: building rapport. Ideally, you do like your users — that includes talking to them.

Good review management takes it all together and adds a reliable system to it. What makes this system is a different question — one we aim to address in this guide.

This guide will equip you with everything you need to know about review management. Basics and theory, intermediate tactics, and niche strategies. It's not as hard as it looks, so long as you're using the right tools for the job.

How does review management work?

What is the difference between a rating and a review?

Quite simple: they are one and the same!
An app rating, across just about any store, contains a rating of 1 to 5, and some text attached to it. A 5-star rating with a thumbs-up emoji and someone saying this app is good is not much of an indicator of what's going on, sweet as it is.

A review, for all intents and purposes, is a rating with some textual meaning attached to it. Even if it's ToS violating, even if it's a long read, and even if it's someone telling the world how much they enjoy a rival app.

What's the point of separating? Because you can read the tea leaves with one, and not the other. Analyzing reviews can yield some good insights: within the last 100 reviews submitted, 26 say you have an annoying UI bug that only pops up on the Galaxy Fold (how rich are your users?), then it's worth looking into.

We could back all of this up with a healthy dose of statistics and charts, such as: **apps with a rating of 4+ get the lion's share of downloads, up to 80%.** Dipping to 3.9 slashes your chances at gaining new business quite dramatically, which is your standard consumer psychology at work. Our primate brains worship big numbers.

3.9 = small number

4 = BIG NUMBER!

Now, how can you tell that investing time and resources into review management is actually worth something? Why do it? Why even try to automate it? Surely a good product with a good page and a well-made app will always sell itself.

Yes, it will. To an extent. You may have a good time selling it until something breaks, and everybody hates your app with a burning passion. Then, suddenly, you dip below 3.9 stars…and that's it. Review management is instrumental in damage control of any magnitude, too.

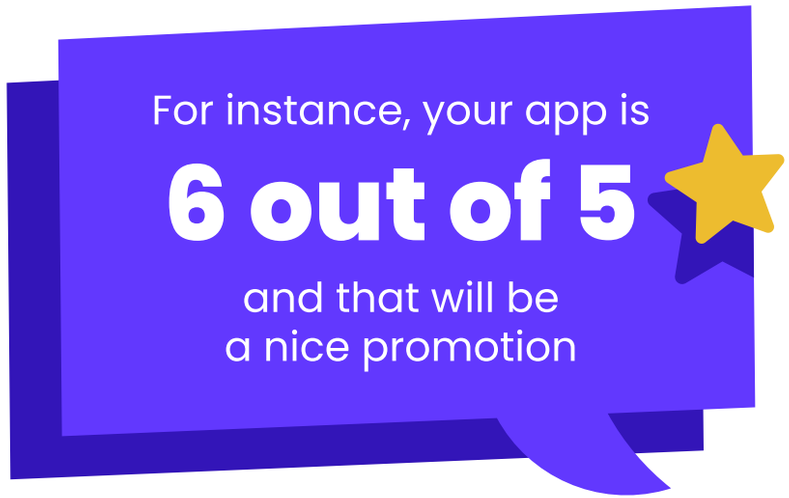
Impact on store rating and sales

Manage reviews well, get tons of good sentiment and improvement in ratings. Better ratings, better sales. But what’s the impact on each particular team you have? How does it all fit together?

Marketing teams

For marketing, the review section is quite limited in terms of use. You could, of course, try to use bulk replies (AppFollow exclusive!) for responding to every single review with some sort of promotion and stuff like that. We would discourage you from doing that.

There is one big thing that you can use the review section itself for: since reviews are social proof, you can absolutely screenshot and use people's reviews as long as they are not using any identifiable information, only a nickname or part of a name to promote your app.



Support teams

For support teams, the impact is very obvious. People will be complaining about your app all the time when something breaks, and it will inevitably break. People will come to you with questions, bugs, inconsistencies, suggestions, and many other things that you will need to work with.

Sometimes, if you have more than one support channel, it's not just the review section, but also email or chat. All of that will have to go to a ticket management system such as Zendesk or HelpDesk.

All feedback should come to a single system used to manage queries. This will help you stay on top of every single problem that may occur, quantify them, and respond fast enough. That also works for categorizing issues, because sometimes, smaller issues just go under the radar, but you need to know about them nonetheless.

Product teams

For the product teams, the review section is probably one of the best places to find product insights that will help the app become better. People will actively tell you what exactly they wish to see. Sometimes it's going to be wishful thinking, sometimes it's going to be something unrealistic, but most of the time, users will tell you exactly what will make the app better in a way that will also sell better.

You could easily categorize this sort of feedback automatically if there's a lot of feedback coming through. You could get summaries and understand what's going on immediately. Say, after running a new update, you could gauge interest and sentiment towards that and work with that later. There is no better place for aggregated user feedback on your app than the review section.

Comparison to app store consoles

It will be absolutely fair to say that one could absolutely go without tools such as AppFollow if your operation is very small and you can respond to the reviews via the app stores themselves. Google Play, App Store, and Samsung Galaxy Store all have their own tools to enable developers to respond to app reviews.

The thing is quite simple, normally: you can get some normal analytics, but you don't get tagging, you don't get any sort of analyzing sentiment, and obviously, you have to do everything by hand.

That might work if you are dealing with maybe one, five, or ten reviews a week. Sure, you will find time to respond to people if you think that is needed at that stage.

That is the most important stage imaginable for review management. The sooner you get five stars or 4.9 stars, the easier it will be to keep them. If you respond to every single review, and besides that, it's your early adopters, you need to make sure they're happy.

What app stores provide

Google Play Console

Google gives you the basics. You can see your reviews, respond to them one by one, and get some simple charts showing rating distribution over time. That's about it.
No categorization, no sentiment analysis, no bulk actions. If someone leaves a review in Korean and you don't speak Korean, good luck figuring out what they're saying.

The response interface is clunky. You have to click into each review individually, type your response, and send it. Multiply that by 50 reviews a day, and you've got a full-time job just keeping up with responses. Plus, Google doesn't tell you if your response helped or if the user changed their rating afterward.

App Store Connect

Apple's approach is even more limited. You get your reviews, you can respond to them, and that's pretty much the end of the story. The analytics are bare bones.
No tagging system, no way to track which responses work, no automation whatsoever.

Apple also doesn't show you reviews from all countries in one place. Want to see what German users are saying? Better switch your region settings and hunt around. Managing a global app becomes a nightmare of switching between different regional views.

Here's what you can't do with native app store tools:

- **No bulk operations.**
  Got 200 positive reviews that need a thank you? Hope you like clicking the same buttons 200 times.
- **No sentiment tracking.**
  Was that 3-star review actually positive or negative? You'll have to read every single one to find out.
- **No performance metrics.**
  Did responding to that angry user help? You'll never know because there's no tracking.
- **No cross-platform view.**
  Managing apps on multiple stores means jumping between different interfaces with different data.
- **No automation.**
  Every response requires manual typing. Every categorization requires manual tagging.
- **No translation.**
  Reviews in foreign languages stay foreign unless you copy-paste them into Google Translate.

> “App store consoles give you review management the way a Swiss Army knife gives you a screwdriver. Sure, it technically works, but nobody's building anything serious with it.
> In competitive gaming markets, every unanswered complaint is a user gifted to your competitor on a silver platter.”



Dimka Plotnikov
Product and Growth Manager at AppFollow

Why centralization matters

Tools like AppFollow possess a lot more functionality that you will need if you want your rating to stay high, such as templates, automation, AI, etc.

Of course, there is the problem of paying extra for the privilege of using these tools as you see fit, and maybe it will take some time getting used to the idea that you need it, but you need it as soon as possible.

Otherwise, you're jumping between stores, you are not categorizing information in the same way, there is a lot of time loss, and by extension, if you want to keep your users happy and maintain a high rating, then you will just need to hire more people, which is expensive.

You need to train, manage them, and they will need to take time to check every single review, or at the very minimum:

- Critical reviews where human input is necessary
- Reviews that violate the terms of service
- Responding to five-star reviews (the easiest to automate)
- All sorts of reviews that are just as valuable as those that you really want to see and respond to personally

Review types — which need more attention?

Reviews come in all sorts of shapes, forms, and flavors. Some will be more important than others, some will be easier to automate, and some will just become an endless headache, and there is no escaping them.

Below, we will outline the types of general reviews that you will encounter and how to assess them in terms of priority, and perhaps what could be a reasonable thing to do if you are to enhance your workflow when it comes to these reviews.

> "The users who write novels in your review section are doing free QA testing. They care enough to document every bug, suggest every feature, and explain exactly why your UI makes them want to throw their phone. Most companies pay consultants for insights this specific."

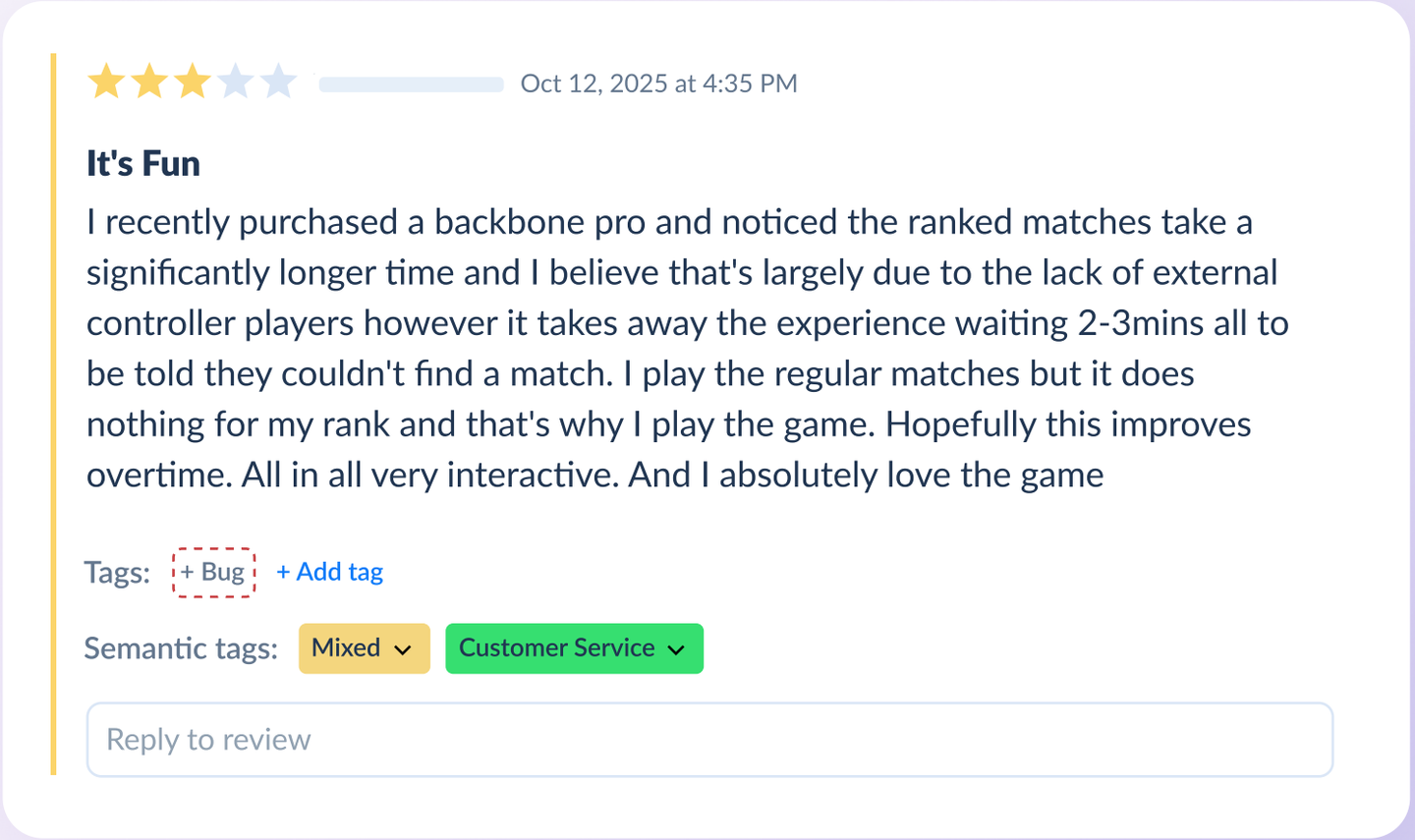


Ilya Kataev
Professional Services Team Lead at AppFollow

Long reads

This is the kind of review that the user has spent quite a lot of time preparing and typing. It's a high-effort endeavor for the user. This could be something positive as they explain their story and why they're using this app and why it's so amazing. This could be a negative situation where they are explaining in detail why your app is failing in so many different ways, and perhaps they're even offering potential ways of solving that.

That kind of feedback is extremely valuable because who else can tell you these things if not your target audience? We might think that we know how things are supposed to be and how valuable they are in terms of features.



PRIORITY: HIGH

Recommended steps: tag and assign a human for a personal response, if possible. If not, provide an automated response to either address the situation as much as you can or offer to continue the conversation later once you have more details.



PRIORITY: CRITICAL

Recommended steps: auto tags (more on these below), templates for common issues, AI replies if there is not enough information to provide categorization, and, of course, human intervention for difficult cases.

Use every tool at your disposal to cover these reviews ASAP. That includes automation, translation for reviews in different languages, AI replies, templates, auto tags, and anything you can lay your hands on to make sure you respond as fast as and as thoroughly as possible.

Negative reviews

Based on feedback from our customers, this is the type of review that gets the most attention, because of course it does. Someone is actively saying how bad your app is, and obviously, you need to respond to this immediately.

At times, users will voice their problems related to how well your app performs, which is absolutely fair, and you can work with that. Sometimes people will be unhappy with your pricing strategy, which will probably not be as easily fixable, at least not directly. Sometimes people will complain about things that are completely unrelated to your app.

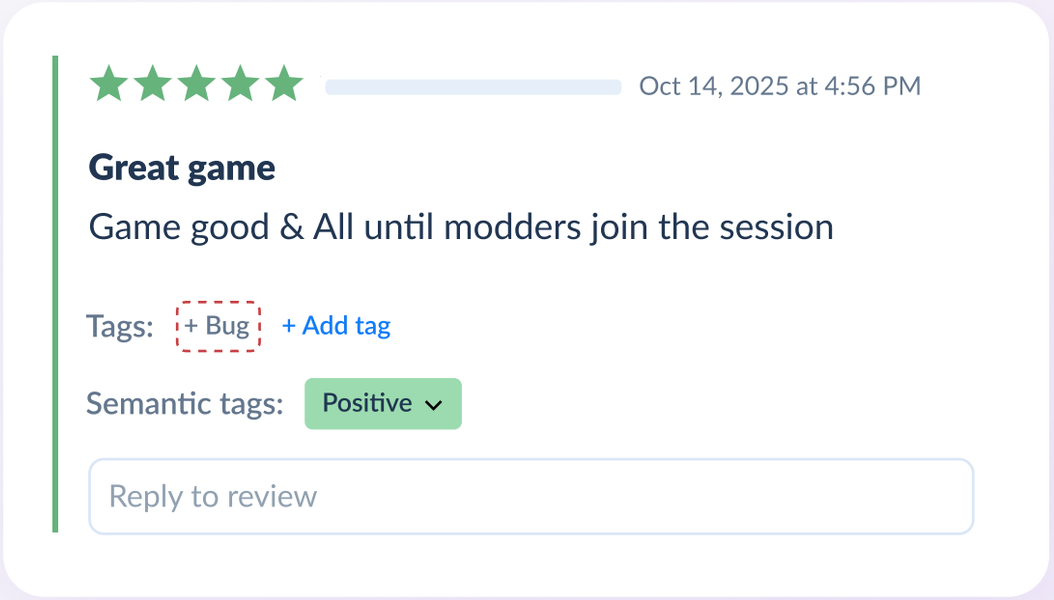
For instance, if it's an online game, people can tell you that someone else is harassing them, and all of that's not the app's fault. They're unhappy with the game itself, and perhaps they're just venting their frustrations in your review section.

Then, bugs, suggestions, and anything else might come too.

Positive reviews

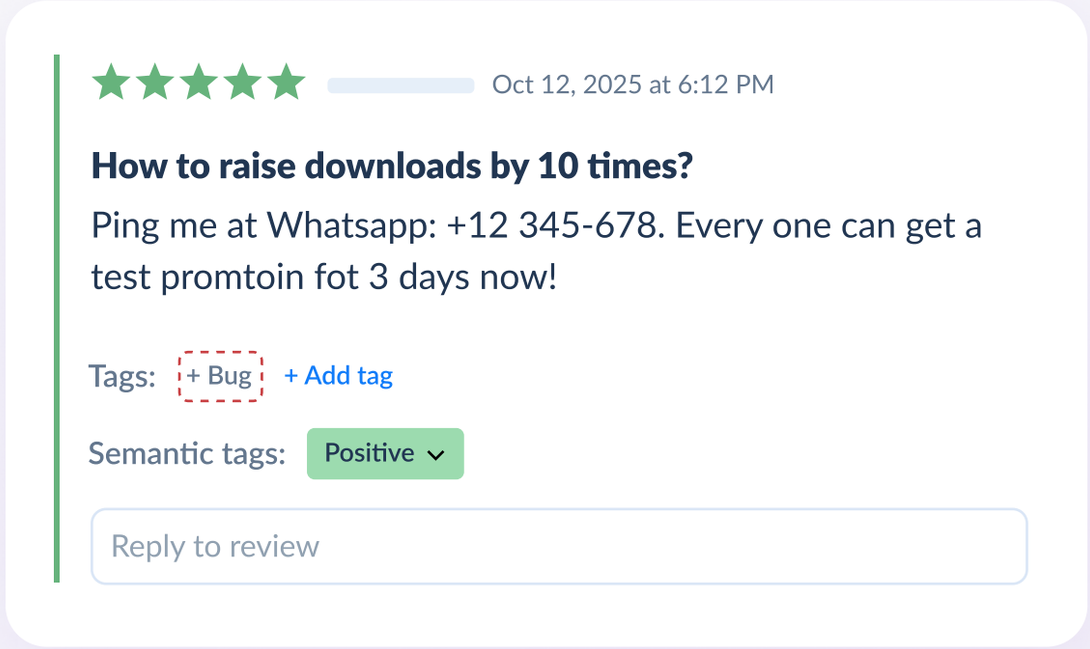
The best sort of review an app maker can hope for. The sort of review that we all want to get, but we will not get them as many times as we would like to. These are exactly the type of reviews that make your rating so nice and sales so high.

You must keep your happy users happy. While it may seem like you don't really need to do that, because they are already happy — “five stars!”, “best app ever!”, you still need to at least let them know that they are appreciated, and you thank them for their stellar feedback. This is easily automatable.



PRIORITY: LOW

Recommended steps: use automation and templates to respond to positive reviews in a variety of ways and thank your users for their patronage. Just in case, run sentiment analysis to make sure that there are no hidden valuable gems or suggestions hiding in your positive reviews.



PRIORITY: MEDIUM

Recommended steps: auto tagging, categorization, and automatic flagging for terms of service violation review removal. A feature in AppFollow that you can use immediately.

Spam and ToS-violating reviews

Yes, the situations where someone is trying to spam your review section with unspeakable, terrible texts that they conjure from the depths of their souls. What a nasty piece of business.

Luckily for you, some reviews are so bad, so out of line, that you can tell the App Store or Google Play, or any other store, to remove them.

It's a little bit difficult to do, it takes a little bit of time for the store to process your request to remove it, especially when someone is very angry and is going off with a whole variety of forbidden words or using very upsetting language. We don't really want our agents or whoever peruses the review section to see that, right?

All of this can be automated as well, and we highly recommend using this use case as one that can be quite easily automated.

Feature requests

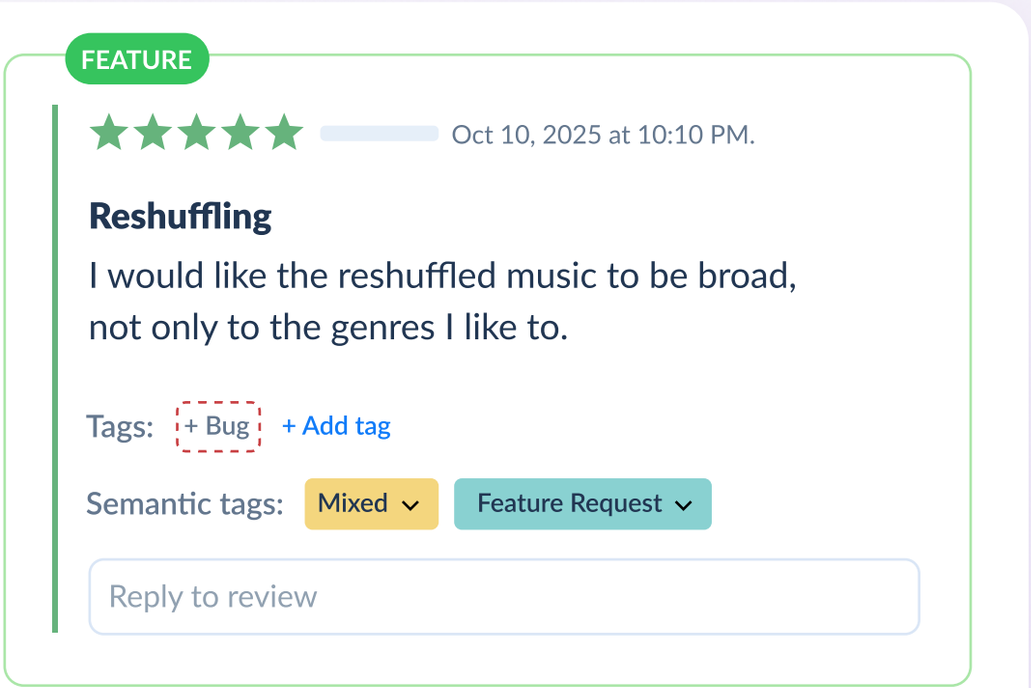
This sort of review is not as valuable for the star rating as much as it is for the information provided within the review. Users will absolutely tell you what they want specifically.

They will request things that they would like to appear in your app. Some will ask for smaller things, some will ask for bigger things. It doesn't matter. What matters is that you want to know about this, and you want to know how many users are asking for a certain feature.

How many of them are doing it now? How many of them have been doing that for a while? Basically, that needs to be a data-driven decision (or at least a data-augmented decision).

With feature requests, there is no way to categorize them under a single star rating. It will be difficult to understand that this is, in fact, a feature request, but drawing any other conclusion from it, such as all the feature requests are three-star ratings or all the feature requests are long reads, will not work.

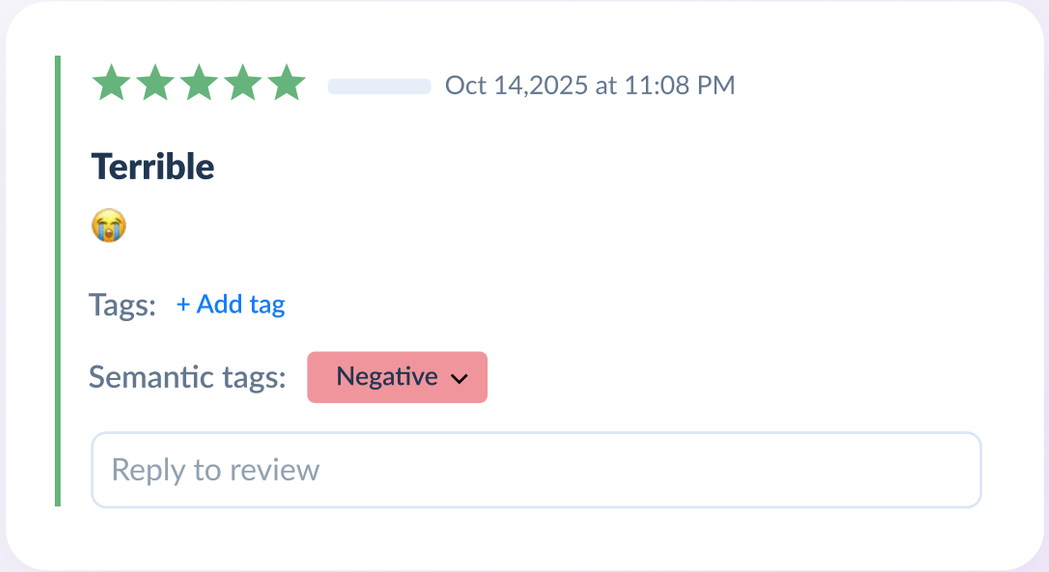
So you will need to look for particular keywords and let automation and, more importantly, AI figure out that someone is indeed telling you what they want. This is the best use case for AI automation when it comes to product teams.



PRIORITY: MEDIUM

Recommended steps: use auto tagging, semantic analysis, AI summaries, and analytics to properly assess how many people want a particular feature. More importantly, with the help of AI you can also gauge why.

If people are saying why they would like that, bring it up with your product teams, and then switch to development if a lot of people are asking for smaller things, because the low-hanging fruit is so very easy to pick.



PRIORITY: HIGH

Recommended steps: AI summaries, semantic analysis, auto tagging, and hope for the best. Basically, what you need to do is figure out that there is indeed a pattern if this is happening.

At the moment, you will probably know if you get a review spike immediately because that's how it normally happens, but if it doesn't, then you just want to keep an eye on positive reviews and analyze whether this positive review is indeed positive and praises your app instead of saying the opposite.

Positive-negative reviews

This is the sort of review you normally see when people are trying to brigade your app rating. A lot of people go to a social platform like Reddit or Discord and decide that the app is the devil and go tank its rating together in a single day.

That happened to Genshin Impact, that happened to Duolingo when AI management was announced, and many other things happen in the same way.

Positive-negative review is a five-star review with very creative negative feedback telling everybody (and don't forget it's a public channel), how bad your app is.

So the rating itself stays fine. Bigger apps can try and tell Google or Apple that their app is under attack and they need to perhaps do something about it. More often than not, the app is bailed out.
Even if they do deserve flak. Smaller apps have to fend for themselves.

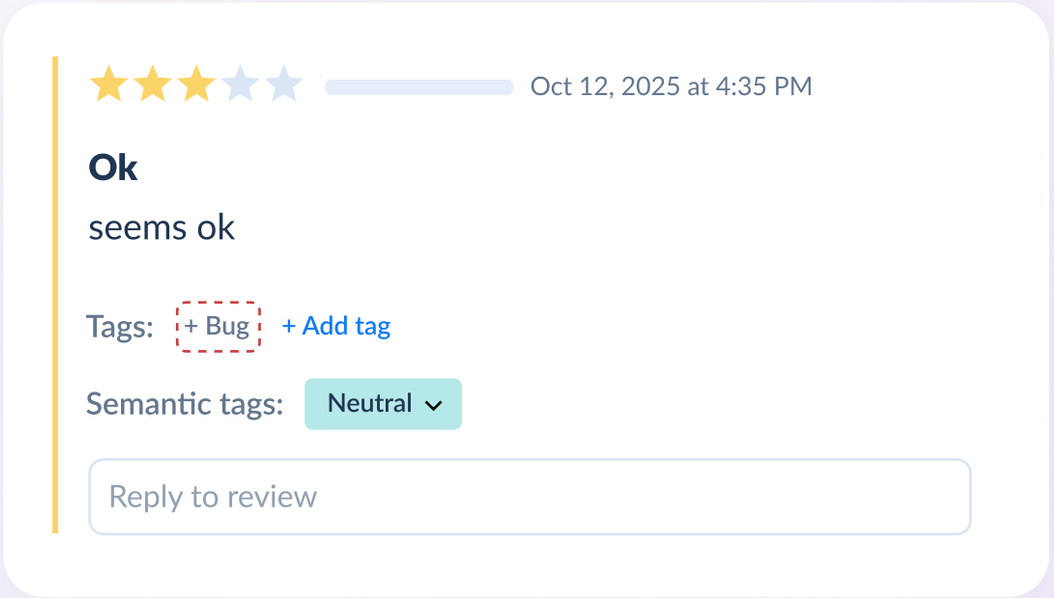
Normally, in times like this, a lot of reviews are removed automatically by Google or Apple in the event of brigading. If it's a 5-star review, you're not so readily protected from this sort of extreme measure, and perhaps you want to know when people are bashing you in positive reviews because you don't want to skip that if it's done on a smaller scale.

Neutral reviews

Some reviews will be about nothing in particular. Say someone says: “well, it's a decent app that does what it says on the tin, and that's about it”, and maybe gives you a three-star rating. What do you do with that?

Obviously, this is good grounds for making sure that this review can be improved. You can respond to the user and try and get them to tell you more about why they think it's just decent and nothing else. Why are they so impartial about it? Maybe they need just a little bit of your love and your attention, and then the magic will happen, and they will tell you what's actually bothering them.

The attention from your team itself will be enough to steer their rating from a neutral 3-star to maybe a more positive 4-star. And that matters. That really does matter.



PRIORITY: LOW

Recommended steps: categorization, templates, and sentiment analysis. Really, that's all you need to get started with letting people tell you more.

You just want to initiate this conversation, and we suggest that you write your templates with the help of AI or perhaps just create a lot of them and let people talk to you. Then, depending on what they say, assign a human, and that might turn into a different type of review — like a long read. Take it from there.

Metrics — what to track?

So we've covered what sort of monsters we meet in the night. What about measuring them top to bottom? How do we do that?

Here's the thing about metrics: **you can track everything, but that doesn't mean you should.** Focus on the ones that move the needle. Some metrics are basic, and everyone should watch them. Others are advanced and help you optimize. And then there are vanity metrics that make you feel good but don't really mean anything.

The key is knowing which is which. Track too few and you're blind. Track too many and you'll drown in data without getting any smarter. The metrics below are the ones that matter for review management. They tell you if your efforts are working, where you're wasting time, and what needs fixing.

Metrics are only useful if they lead to action. If you're not going to change anything based on what a metric tells you, stop tracking it. Your time is better spent elsewhere. Now let's see which metrics are worth your time when it comes to review management.

"You can track a hundred metrics and still miss the only three that matter: your star rating, user satisfaction (whichever you prefer), and reply effect. Get these right and the rest becomes academic. Get them wrong and no amount of data visualization will save you."



Janire Rodríguez Indias
Head of Customer Success and Professional Services at AppFollow

Response speed

How fast you respond to reviews matters more than you think. Users expect quick responses, especially when they're angry. A response within 24 hours is good. Within 8 hours is better. Within 2 hours and you're a hero.

But speed isn't everything. A fast, terrible response is worse than a slow, thoughtful one. The goal is to be both quick and helpful. This metric tracks your average time from when a review is posted to when you respond.

Track this separately for different types of reviews. Negative reviews should get faster responses than positive ones. Critical bugs need immediate attention. Five-star thank yous can wait a bit.

Use this metric to set targets for your team. If your average response time is 48 hours and your target is 12 hours, you know you need more people or better processes.

Response rate

What percentage of reviews are you actually responding to? This metric tells you how much of your review volume you're covering. Aiming for 100% is unrealistic unless you're tiny. But 80% is achievable with the right tools and processes.

Different review types need different response rates. You should respond to 100% of negative reviews and long reads. Positive reviews can be lower priority. Terms of service violations might not need responses at all.

Track this by review type, by team member, and over time. If your response rate is dropping, you either need more automation or more people. If it's too high, you might be wasting time on reviews that don't need responses.

This metric helps you understand capacity. If you're only responding to 40% of reviews, you're missing opportunities to improve ratings and gather feedback.

Average rating

Your app's overall grade point average. It's every rating from every user, crammed into one number. No words, just stars. This is the first thing potential downloaders see. High average means they're in. Low means they're ghosting you faster than a bad date.

Track this daily and watch for trends. A sudden drop means something broke or users are angry about an update. A steady climb means your improvements are working. Small changes matter here. Going from 3.9 to 4.0 can dramatically increase downloads.

Don't just track the overall number. Look at the rating distribution too. Are you getting mostly 5s and 1s? That suggests polarization. Mostly 3s and 4s? You're probably boring but functional.

This metric connects directly to revenue. Apps with 4+ stars get the lion's share of downloads. Below 3.9 and you're in trouble.

Reply effect

The reply effect shows if your responses actually change user ratings. When you reply to someone's complaint and they change their rating, that's the reply effect in action. This metric is only calculated if the change happens within three months.

Here's what counts: a user responds to your reply and boosts their rating; that's a win. They drop their rating after your response; you could do better. They change their review without replying to you, doesn't count toward reply effect.

You can track automation reply effect versus manual reply effect, depending on who left the review response, AI or a human. Automation might handle volume, but humans often get better results for complex issues.

Now here's the interesting bit about the 5-star review reply effect: it can only be negative. How can you get a positive change to the review rating if it's already 5 stars? You can't go higher than 5.

So when you respond to 5-star reviews, you risk them changing their mind and dropping the rating. This is why many teams focus reply efforts on 1-4 star reviews where there's room for improvement.

Net Promoter Score
(NPS)

"Would you recommend us?" It's about finding your cheerleaders and your naysayers. A high NPS means you're the cool app, with users likely to stick around and bring friends. Low NPS means it's time for a strategy update.

The formula is simple: percentage of promoters (9-10 ratings) minus percentage of detractors (0-6 ratings). The score ranges from negative 100 to positive 100. Anything above 0 is decent. Above 50 is excellent. Above 70 is world-class.

NPS tells you about loyalty and word-of-mouth potential. Users might tolerate your app but not recommend it. Or they might love it enough to evangelize. This metric captures that difference. Use NPS surveys in your app or calculate it from review ratings. Track it over time and after major releases to see if changes improve user advocacy.

Customer Satisfaction Score
(CSAT)

CSAT is the smiley face on your app's report card. It's a straight-up ask: "Hey, you happy with what we're doing?" This score tells you how many thumbs are pointing up.

Calculate it as the percentage of satisfied customers (4-5 star ratings) out of total respondents. Unlike NPS, CSAT focuses on current satisfaction rather than likelihood to recommend.

CSAT is useful for measuring immediate reactions to updates, new features, or support interactions. It's more tactical than NPS. Use it to gauge short term satisfaction while NPS tracks long-term loyalty.

You can track CSAT for specific features, support interactions, or overall app experience. It's simpler than NPS and easier for users to understand.

Sentiment score

This one's the mood ring for your app. It takes all those reviews, analyzes the text, and tells you if the vibe is more love festival or angst-ridden diary. A high sentiment score means your app's users are happy. It's a direct reflection of how well you're meeting expectations.

Sentiment analysis looks beyond star ratings to understand what users actually say. A 4-star review saying "this app is terrible but I have to use it" has a different sentiment than a 4-star review saying "love it, just needs one small fix."

Track sentiment by review type, time period, and app version. Watch for sentiment drops after updates or spikes during outages. This metric often predicts rating changes before they show up in your average.

Use sentiment analysis to prioritize which reviews need human attention. High-volume negative sentiment reviews need immediate response.

Agent performance

At the heart of every team, there are metrics to track. Otherwise, how can you tell if you're making progress? Agent performance shows you how your support team is doing with replies and review management.

Track reply counts, average reply time, and reply effect for each team member. See how fast they respond and whether their responses actually help. Some agents might be fast but ineffective. Others might be slow but get great results.

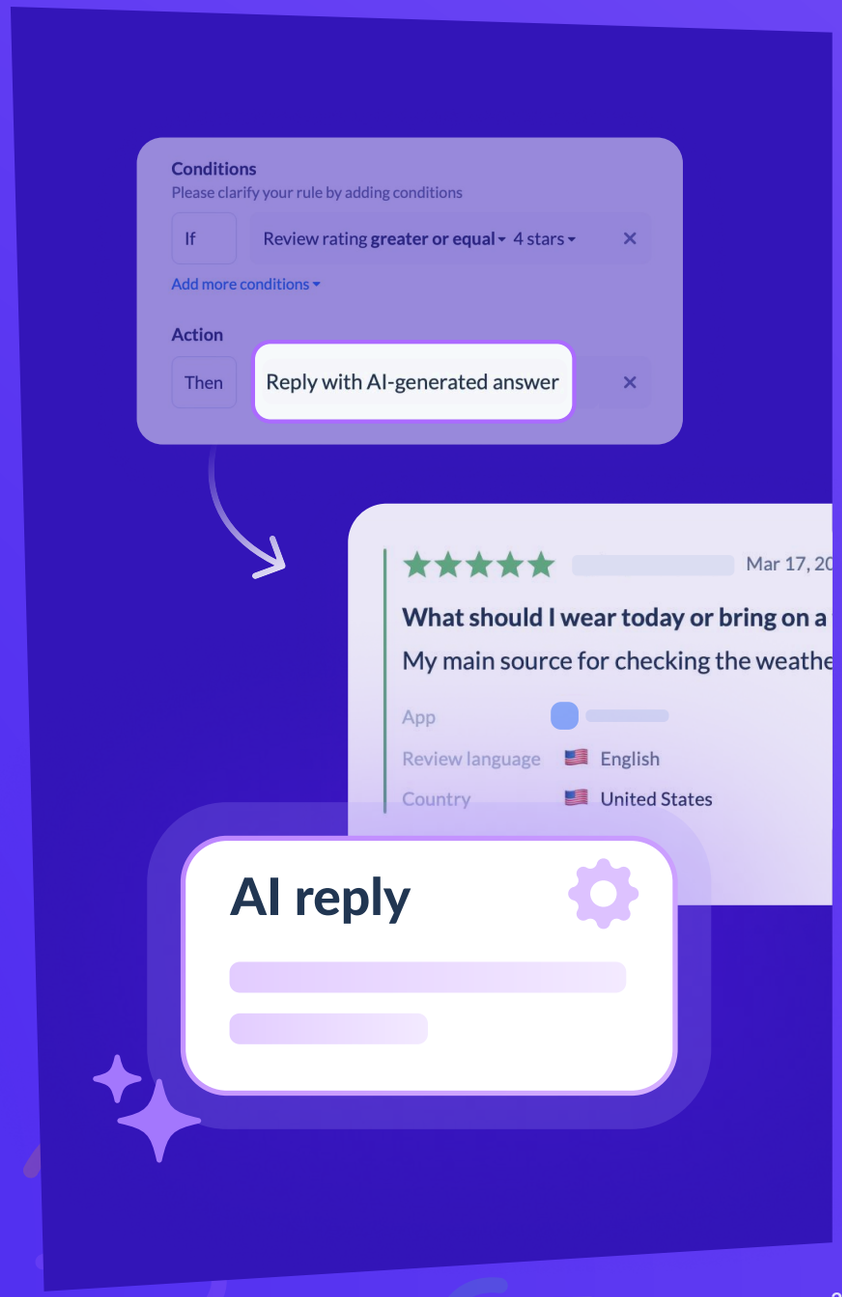
Monitor tickets closed, reviews tagged, and customer satisfaction scores by agent. This helps with training, workload distribution, and performance reviews. You can also compare automation performance to human performance to see where each works best.

Don't just track individual performance. Look at team trends, seasonal patterns, and how performance changes with workload. Use this data to improve processes and training.

AppFollow best practices

Now that you fully understand what needs to be done to make sure that your app stays highly rated and all of your users remain happy, and your product teams are full of insights they want to use all the time, let's talk about how to do it.

Below you'll find workflows that you can use in particular scenarios where doing it by hand won't quite cut it. Let's start with understanding what's going on, and for this you will need AppFollow analytics. The way it works is that you get auto tags, summaries, sentiment analysis, and some other bits and bobs.



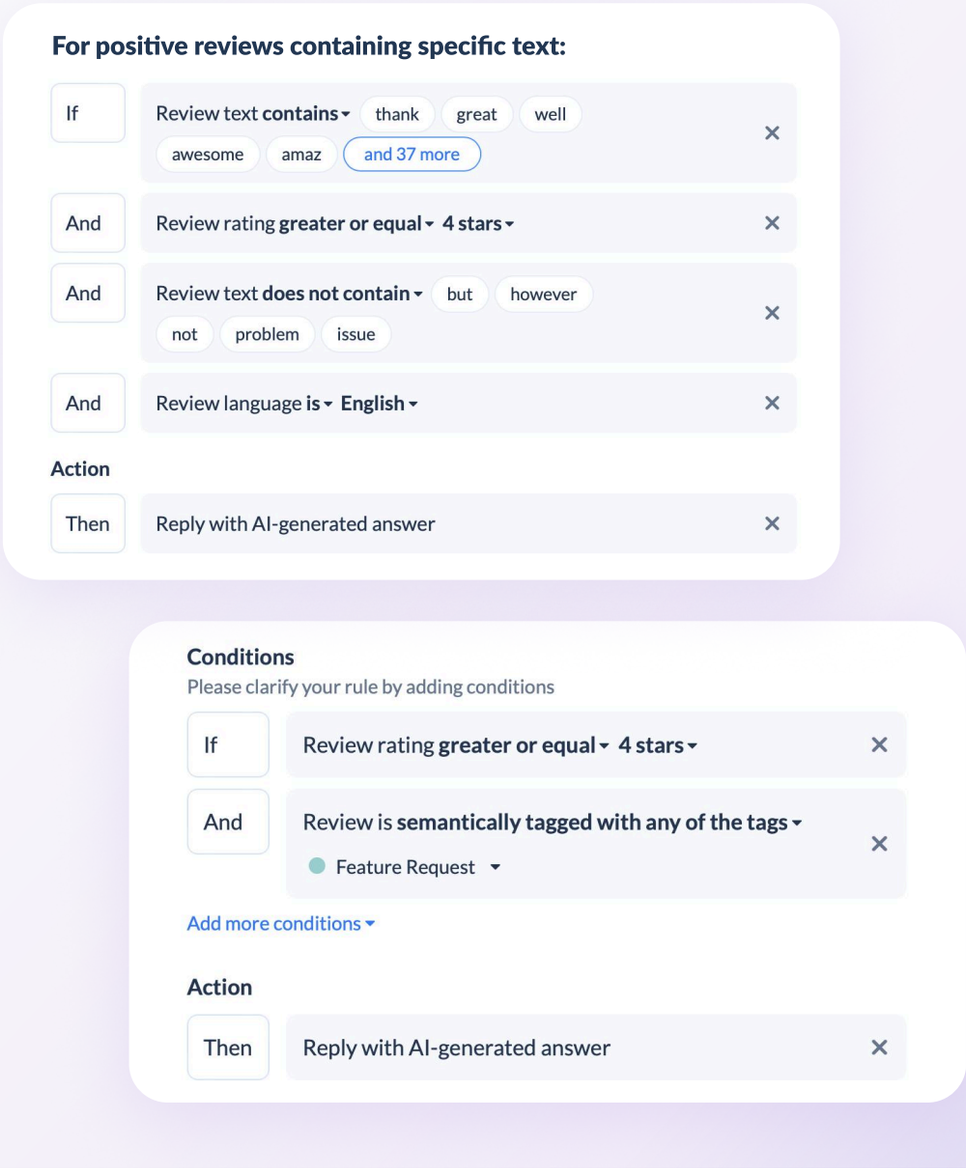
Auto replies

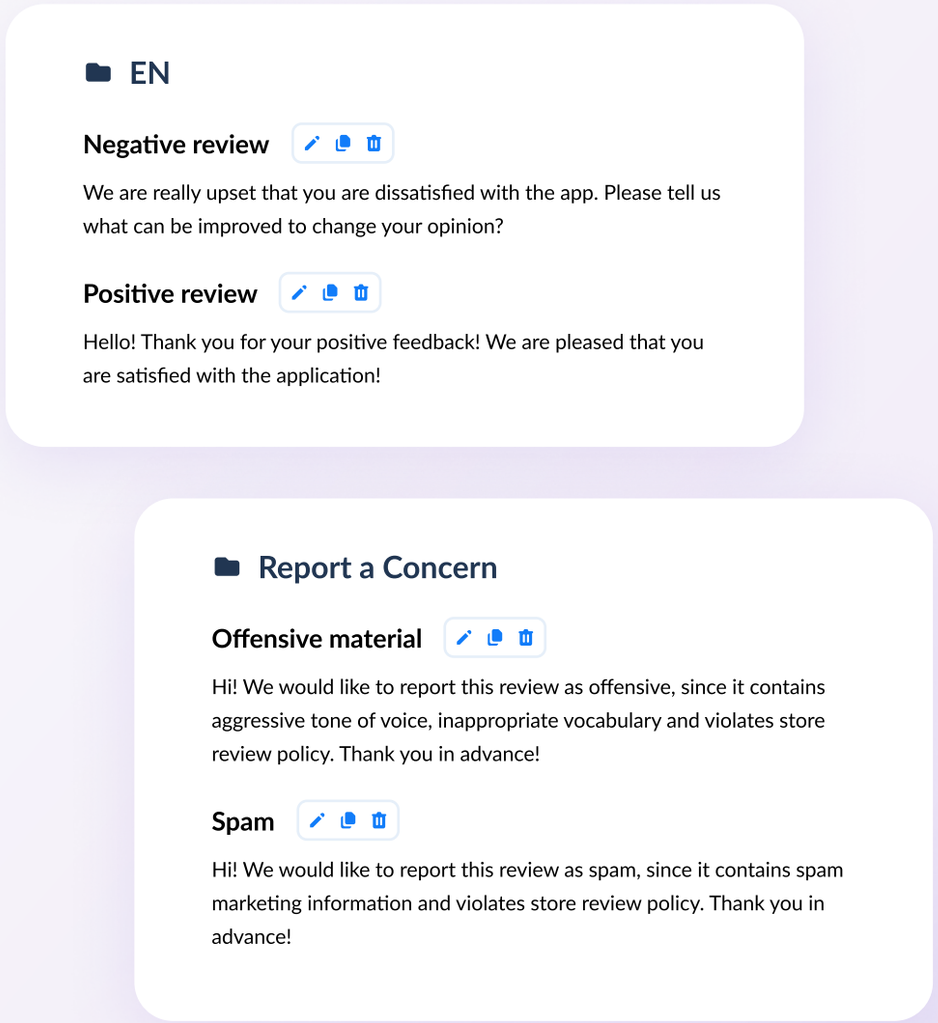
The first step towards greatness! Set up some templates, and then let the robots categorize feedback via auto-tags and automatically respond based on the rules you've set up. Mind you, there is a difference between automation and AI. Automation is a set of pre-determined rules, where if something outside of your framework happens, the system won't do anything.

Auto-replies are your lifeline when dealing with an avalanche of similar reviews. Forget manually responding to each one, let AppFollow do the heavy lifting. You can have the system use random templates for a particular condition, say your own varied templates for the same occasion, and auto-translate your templates into any language.

Conditions for auto-replies feature keywords, exact matches, star ratings, character counts, etc. You can also set up automation based on language, translation, sentiment, and featured status. Start with your 5-star reviews, let automation handle the praise while your team tackles complaints and feature requests.

Use auto-replies for the most repetitive scenarios: thank you messages for positive reviews, acknowledgment of bug reports, requests for more information on vague complaints. The key is to set clear rules and monitor performance to make sure the automation is helping, not hurting.





Templates

Templates are the building blocks of efficient review management. Create different templates for different scenarios: positive reviews, negative reviews, bug reports, feature requests, pricing complaints. The more specific your templates, the more personal they'll feel to users.

But here's the trick: don't make them sound robotic. Write templates that feel human, then create variations so you're not sending identical responses to everyone. Use placeholders for user names, app names, and specific issues when possible.

Organize templates by review type and sentiment. Train your team on when to use which template and when to write custom responses. Templates should handle 80% of your reviews, leaving humans free to tackle the complex stuff.

Keep templates updated. When you launch new features or fix bugs, update your templates to reflect current information. Outdated templates that reference old app versions or missing features make you look out of touch

Create templates in multiple languages if you have international users. Don't rely on machine translation for customer-facing communications. Get native speakers to write proper templates for your key markets.

Auto-tags

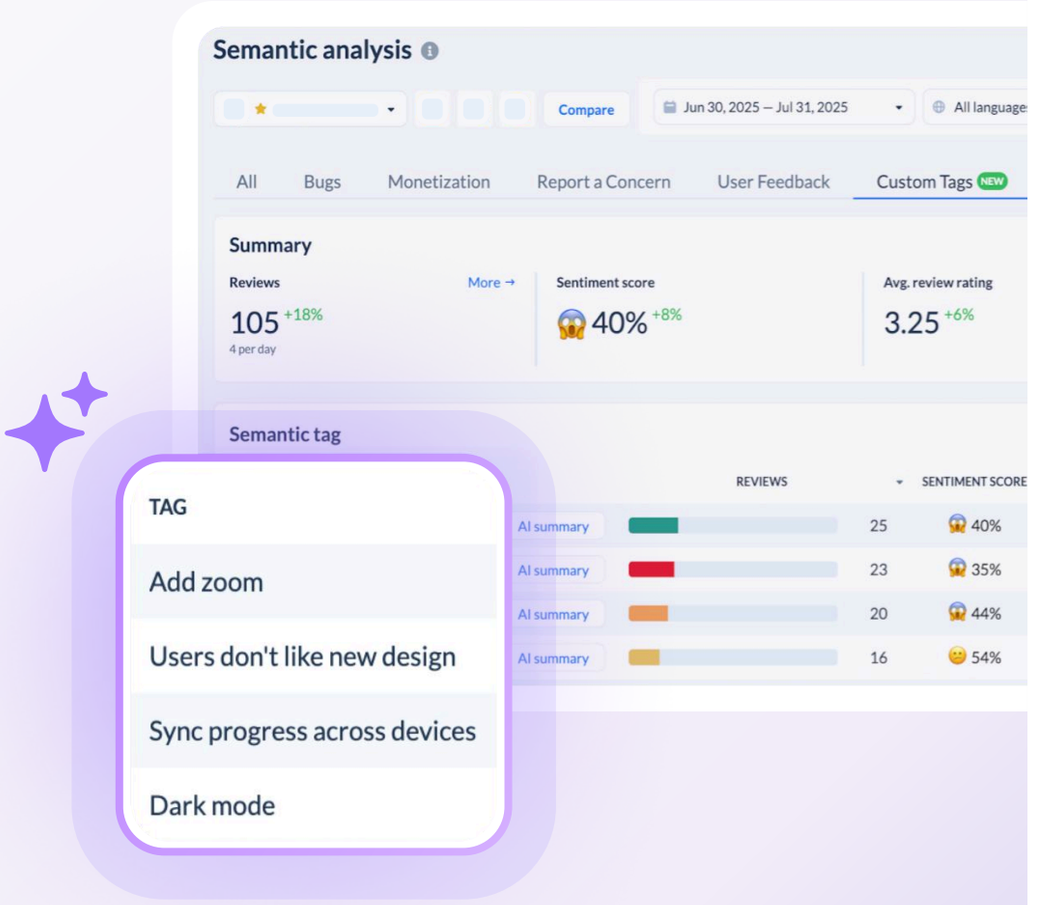
This feature will help you automatically look for particular keywords in your reviews and then automatically apply tags, hence the name. We recommend using this feature for all of your review management endeavors.

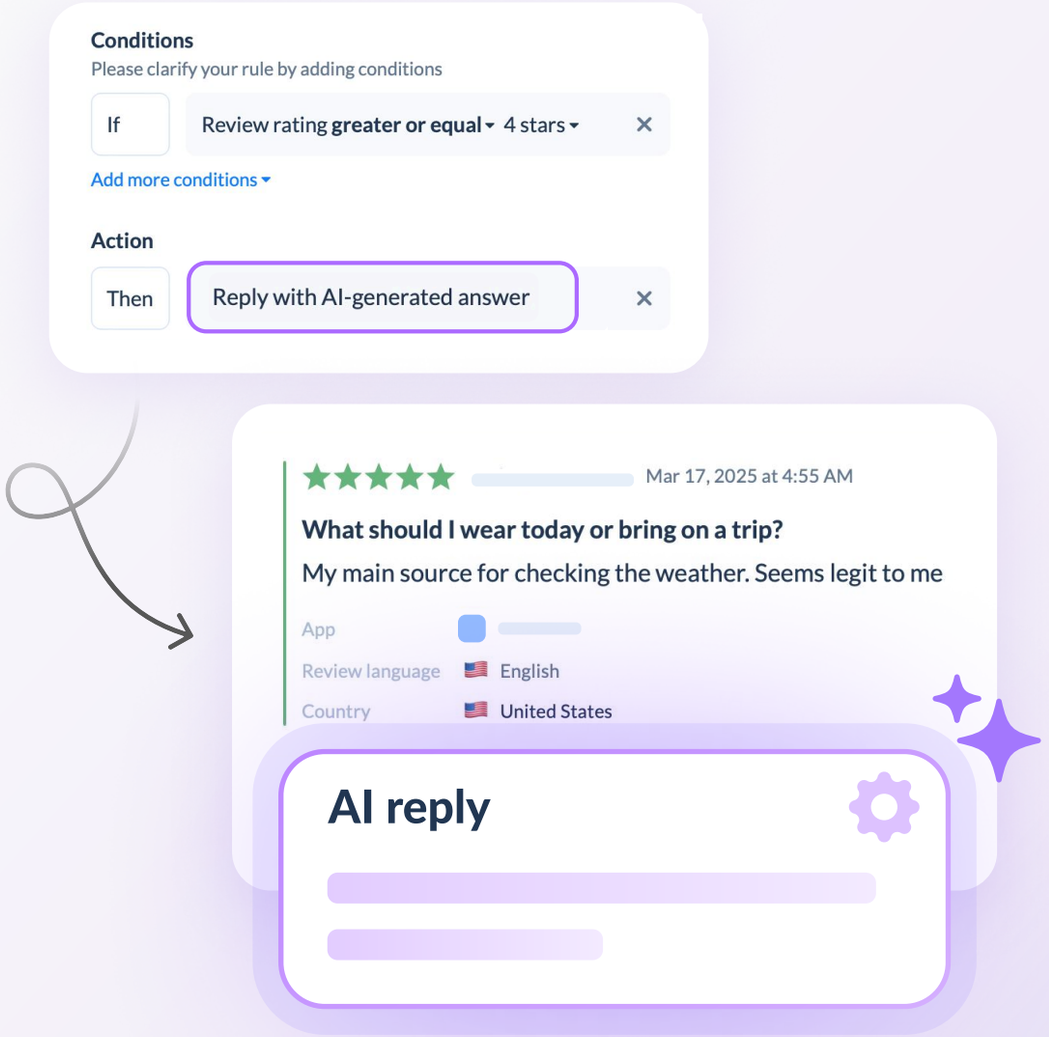
Auto-tags work by scanning review text for specific keywords and phrases, then applying relevant tags automatically. Set up tags for common issues: bugs, crashes, payment problems, feature requests, UI complaints. This lets you quickly see patterns in your feedback.

Use auto-tags for all types of reviews, particularly negative ones, long reads, and feature requests. They're useful for support and product teams who need to quickly identify and prioritize issues. The more comprehensive your tagging system, the better insights you'll get.

Start with obvious keywords but refine over time. "Crash" and "freeze" are easy wins for bug tags. "Love" and "amazing" work for positive sentiment. But also look for more subtle patterns like "wish it had" for feature requests or "used to work" for regression bugs.

Don't over-tag. Too many tags make the system useless. Focus on actionable categories that help you prioritize and route reviews to the right teams. A good tagging system should let you answer questions like "how many people complained about login issues this week?"





AI replies

AI takes over based on conditions, but beyond what you can script yourself, and responds to all reviews that qualify. AppFollow's AI automation for app store review replies is fast, smart, and works on both App Store and Google Play. Based on the rules you set up, you can automatically respond to reviews with AI's help, completely hands-free.

You can set the tone and add extra information to the replies with custom instructions. The system only uses public data, your private info stays private. Define your rules using ratings and keywords, choose AI Reply as your action, watch how it performs, and adjust your approach.

To use AI replies manually, jump into "Reply to Reviews" in the left menu. Pick any review that needs attention. Hit "Generate AI reply," tweak the response if needed, and send it out. AI replies help your reputation strategy scale infinitely.

You can even use AI replies to schedule posts in bulk. Update everyone who complained about a bug in bulk, and make every single response unique with a single push of a button. Start conservative with AI replies. Let it handle simple positive reviews first, then gradually expand to more complex scenarios as you gain confidence.

Monitor AI reply performance closely. Check the reply effect to see if AI responses are actually helping or hurting your ratings. Some users can tell when responses are automated and might react negatively.

AI summary

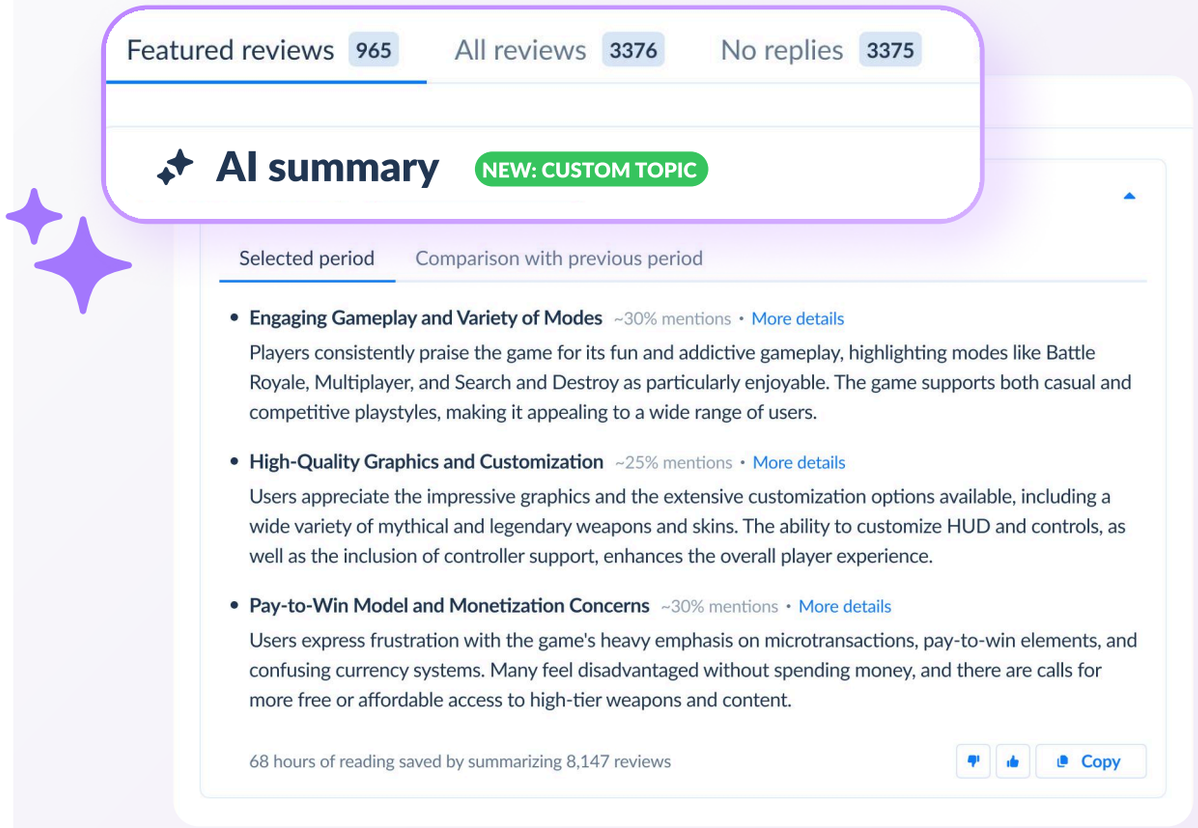
One of the best things you can get for insights at a glance. Having a wild week after releasing a new feature, and you'd like to know what people are thinking at the moment? This is the feature for you. AI Summary makes dealing with customer feedback dead simple.

It scans hundreds of reviews instantly so you can act immediately, instead of spending time analyzing everything first. You can tweak the range of things you'd like to analyze: only positive reviews, or maybe only negatives, specific tags, time periods, whatever you need.

Use AI summary for both semantic analysis and reviews analysis. If you'd like a quick update when something changes, say to your Slack channel, you can set that up too. This is invaluable for staying on top of sentiment changes after app updates or during crises.

AI summary helps product teams understand user feedback at scale. Instead of reading through 500 reviews to understand what users want, get a summary that highlights the key themes and requests. This turns hours of manual work into minutes of insight.

Set up regular AI summaries for your team. Weekly summaries of all feedback, daily summaries during launch periods, and immediate summaries when negative sentiment spikes. The key is making insights actionable and timely.



"Your review section is a town hall meeting that never ends. Hundreds of voices talking over each other about bugs, features, and feelings. Understanding what everyone is talking about is invaluable — to all of your teams, at the same time."



Ilia Kukharev, Product Manager at AppFollow

Conclusion

So here we are. You've made it through the complete guide to review management, from understanding why you should worry about reviews to implementing AI-powered automation that scales infinitely.

The apps that win aren't necessarily the ones with perfect products. They're the ones that listen to their users, respond quickly to feedback, and continuously improve based on what people want.

You now know that 3.9 stars means low sales and 4+ stars means big sales. You understand the different types of reviews and which ones deserve immediate attention versus which ones can wait.

You've learned how to track the metrics that actually matter and ignore the vanity numbers that just make you feel good.

Most importantly, you've discovered that you don't have to do this alone. Tools like AppFollow exist precisely because managing reviews manually doesn't scale. Don't wait until your rating drops below 3.9 and you're scrambling to recover. Don't wait until you have thousands of reviews pouring in daily. Start with your first review, respond to it, and build from there.

The 4.5+
star rating awaits

APPFOLLOW

Never miss user feedback again with AI-powered reputation management

Try AppFollow today

For inquiries, please reach us at sales@appfollow.io

Please avoid sharing the content of this guide without crediting AppFollow. © AppFollow.fi Oy (Finland), 2025