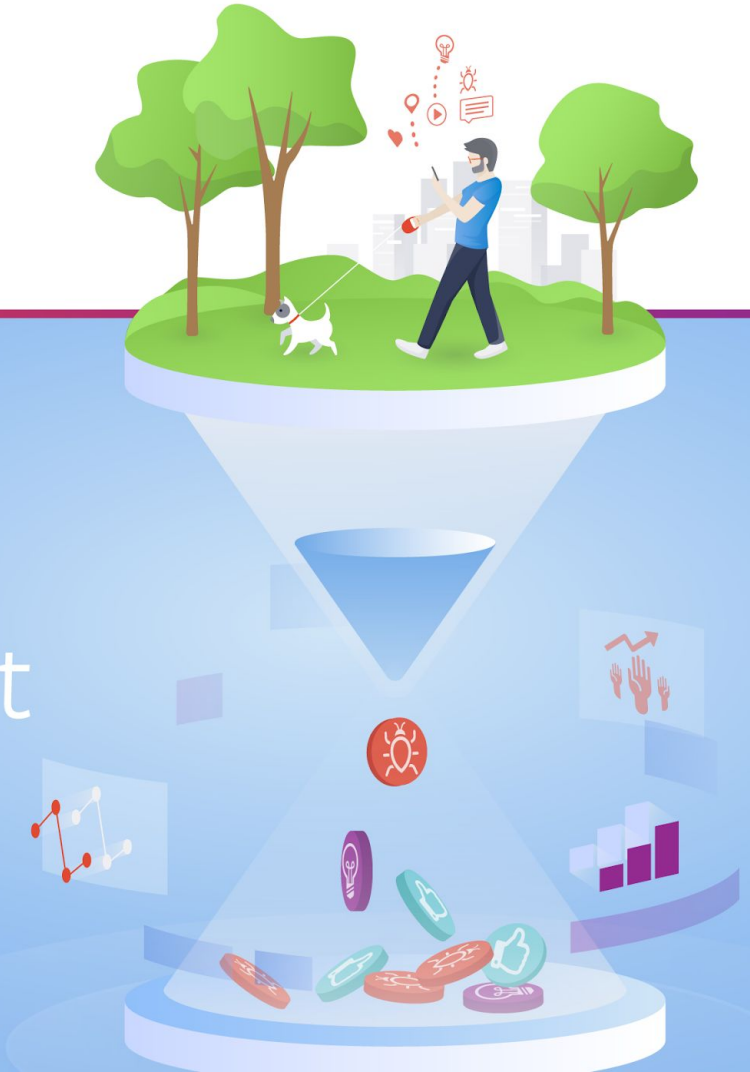
centercode

What You Need to Know:

Beta Test Management



About This Ebook

Chances are good that you've heard the term "beta" before. But when you've been tasked with actually *running* a beta test or managing an entire beta program, the word takes on a whole new (and often nebulous) meaning.

Where do you start recruiting? How many testers do you need? What key metrics should you look at? If you feel like you're going in blind, you're not alone. Many aspects of beta testing are neither intuitive nor straight-forward, and it can be difficult to find reliable resources (that's why Centercode was founded in the first place!)

This ebook is designed to walk you through the basics of beta test management. Inside, you'll find a basic guideline for understanding the goals of beta testing, identifying and resolving core issues, and a step-by-step process for managing your project.

This piece is for you if:

- you're a quality, engineering, or product management professional, or
- you've been tasked with running a beta test or program.

Table of Contents

BETA BASICS 4

- DEFINITION 5
- BENEFITS OF BETA TESTING 9
- MAJOR CHALLENGES 11

TOOLS 14

TESTING PROCESS 21

- PLANNING 21
- RECRUITMENT 26
- KICK-OFF 33
- MANAGEMENT 39
- CLOSURE 47

WRAPPING UP 51

DELTA TESTING SOLUTIONS 52

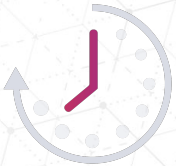
centercode



BETA BASICS

centercode



Definition of a Beta Test

"Targeted users using actual products in real environments over time."

Also Known As

Beta testing goes by many names and may have different applications depending on the company. That said, they all focus on **securing real-world customer feedback before product launch**.

- User Acceptance Testing (UAT)
- Customer Acceptance Testing (CAT)
- Customer Validation (CV)
- Field Trials
- Pre-release testing

Note: While we use the term "beta" throughout this ebook, the same principles, processes, and benefits apply - no matter what you call it.

Beta testing...

- is the final test prior to shipping your product,
- generally begins after the alpha testing phase,
- begins when the product is at or near feature completion,
- includes existing and/or targeted prospective customers,
- commonly prioritizes quality improvement (i.e., squashing bugs),
- offers a final assessment of the complete product before launch, and
- extends QA testing (but should never replace it).

And most importantly, beta testing…

- differs greatly in challenges, execution, and benefits based on tools, product type, target market, beta goals, and development phase.

Applications for Beta Testing

There are many different applications for beta tests, each with its own set of challenges, goals, and benefits. The way you run your beta test will depend on the type of product, its target market, your test goals, and how far along the product is in its development.

This chart illustrates an example of a traditional, performance-based hardware beta test that's aimed at the business market.

Product Type	Target Market	Primary Goal	Test Phase
Software	Consumer	Requirements	Pre-alpha
✔ **Hardware**	Children	✔ **Performance**	Alpha
Video	✔ **Business**	Focused	✔ **Pre-release**
Mobile	Enterprise	Stress	Agile
Web	Government	Awareness	Post-release

centercode

Benefits of Beta Testing

Beta testing provides true validation from users in the wild and improves product quality beyond the inherent constraints of the quality testing phase. This results in:

- Reduced Development Costs
- Increased Sales
- Reduced Support Costs
- Reduced Returns
- Increased Customer Satisfaction
- Improved Product Planning

* Other benefits are unique to product type, market, beta goals, and phase (we'll discuss each one later)
* See the ***Beta Test ROI Kit*** for details on the returns from even a minimal investment in beta testing.

The Beta Message

Everyone in your organization will see the value of beta differently. To gain buy-in, it helps to communicate how beta testing helps each department achieve their unique goals.

Product Management. Beta testing uniquely covers every aspect of a product, including support and documentation.

Marketing. Beta testing is an opportunity to gain initial product impressions from your target market. This feedback can be used to improve your launch outcomes.

Engineering. Beta testing ensures your product is performing exactly as expected in the hands of real customers.

Executive Level. Beta testing provides a high-level view of the entire product. You can then use this information to make crucial decisions prior to launch.

Quality. Beta directly extends your quality efforts within real-world environments that are difficult or impossible to simulate in a test lab.

Sales. Beta testing can generate testimonials and other assets useful in early sales efforts. In addition, beta quality enhancements can lead to higher reviews, resulting in more sales.

centercode

Major Challenges

1. Dedicating Qualified Resources. Beta testing generally happens at the least opportune time, right when the product team is frantically trying to close up every loose end and has little available time or energy to give to beta testing.

2. Recruiting the Right Beta Testers. Not every interested applicant makes a good beta tester for your product. Finding an adequate pool of targeted but unbiased testers is a difficult task.

3. Maintaining User Participation. The single most common challenge in beta is achieving high participation from beta testers. The industry average is lower than 30%, meaning that 70% of testers fail to provide any meaningful feedback.

Major Challenges (cont.)

4. Collecting Useful Feedback. Not all feedback is valuable. Useful feedback is relevant, descriptive, and contextual, and either results in a direct improvement to the quality of your product or meets some other specific goal of your beta test.

5. Organizing and Distributing Feedback. Successful beta tests generate an enormous amount of feedback. Organizing and distributing relevant data to your different stakeholders is a complicated and time-consuming task.

6. Deriving Actionable Recommendations. This is the ultimate goal of any beta test. Without useful feedback provided by engaged members of your target market, how can you hope to form actionable recommendations that your product team can reliably use?

centercode

Top 10
Beta Mistakes

1. Not including your stakeholders during planning
2. Not having a well-defined plan
3. Underestimating ramp-up time requirements
4. Releasing a non-viable product to beta testers
5. Recruiting too few or too many beta testers
6. Setting a test period that is too short or too long
7. Poorly motivating and/or managing testers
8. Losing time and feedback due to ineffective tools
9. Failing to manage feedback effectively
10. Managing incentives poorly

centercode



TOOLS

Finding the Right Tools for Beta

Your tools contribute greatly both to the effectiveness of managing your beta test processes and your ability to deliver useful results to your stakeholders. Here are the biggest benefits of using the right tools.

Shortens beta cycles. Tools that allow you to maintain a pre-profiled pool of candidates reduce your recruiting and ramp-up period substantially, allowing you to achieve more, faster.

Reduces time investment. Tailored tools cut the time you spend coordinating, communicating, and reviewing feedback from your beta test by more than half.

Increases tester participation. Providing a clear path for feedback reduces friction and results in higher participation rates.

Improves feedback quality. Dedicated tools help generate structured feedback and data that is more clear and actionable.

Increases program visibility. Tools that have the ability to integrate with your existing systems ensure your entire team is aware of feedback and able to react accordingly by fixing issues and making other improvements.

centercode

Productivity Tools

Examples
Email, spreadsheets, word processors



- Accessible and inexpensive
- No learning curve



- Doesn't scale (more than ~10 participants is extremely difficult to manage)
- No means of secure collaboration and communication between beta testers
- Not sufficient for a team environment (little to no visibility or accountability)

centercode

Non-Beta Software and Services

Examples
BugZilla, Jira, SurveyMonkey, Get Satisfaction, phpBB, WordPress, etc.



- Inexpensive
- Easily accessible
- Many choices



- Utilizing multiple systems is disjointed for testers (reducing participation)
- Difficult to manage and produces fragmented, disconnected sets of data
- Participation is difficult to track

In-House Beta Systems

Examples
A tool developed within an organization for managing beta tests



- Built for beta, customized for specific processes and needs
- Dedicated experience for testers reduces friction and increases feedback
- May include other custom system integrations



- Immensely expensive to develop and maintain
- The platform doesn't evolve without ongoing investment
- Risks going end-of-life if internal support wanes

Commercial Beta Tools

Examples
Test.io, UserTesting, Validately



- Built for beta (community, agreements, CMS, surveys, feedback, forums, reporting)
- Low learning curve for customers/testers
- Flexibility to meet evolving requirements, integration with existing tools



- Niche offering with few vendor options
- More costly than using existing productivity tools
- New application to adopt

Other Useful Tools for Beta

Project Planning Tools (e.g., Microsoft Project). A beta is a project like any other, with milestones, activities, and resources. Using project planning and task management tools help you stay organized.

Betabound or Other Tester Communities. Centercode's tester panel of 200,000+ profiled users offers access to hundreds of testers whose technographic and demographic qualifications match the needs of your test.

Social Media (e.g., social networks, portals, forums). Mined correctly, the internet offers an endless collection of great beta testers.

The Centercode Resource Library. We've produced a number of educational resources and useful kits to help run successful beta tests, all available in our *resource library*.

centercode



TESTING PROCESS

PLANNING

The Beta Test Plan

Every good project begins with a plan. In the case of beta testing, a plan ensures that your project parameters, goals, milestones, activities, and participant demographics are well-defined and communicated to everyone involved.

We've developed three kits that include everything you need to build a comprehensive beta plan. Each kit includes a complete walkthrough, a detailed sample, and a template to create your own plan.

Download the free kits:







Writing a Beta Test Plan

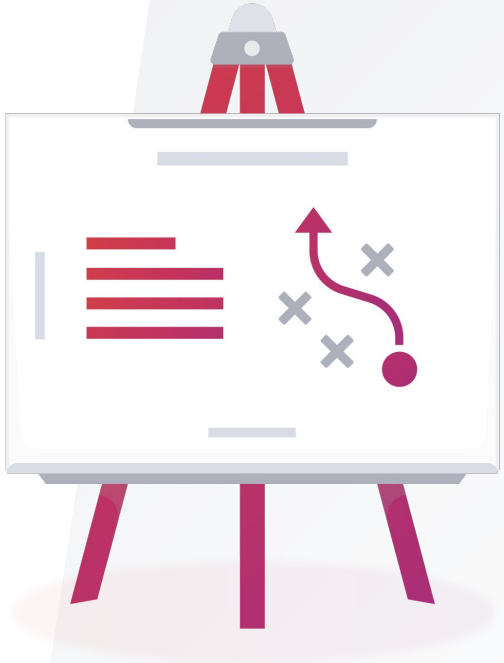
While there are countless ways to structure your beta plan, at the very minimum, any plan should include the following components:

1. **Budget**. Identify any costs that will be associated with your project, including tools, beta units, shipping logistics, and incentives.

2. **Product Definition**. Create an outline or map of your product, including an overview of its key features — especially the ones you need evaluated during your beta test.

3. **Test Parameters**. Outline the basic scope of the project, including total number of beta testers, project schedule, and milestones.

4. **Target Market Definition**. Detail the target market for the product. If the market is segmented, break down the types of users into profile pools. Include the tester requirements (time, hardware, etc.) in this definition.



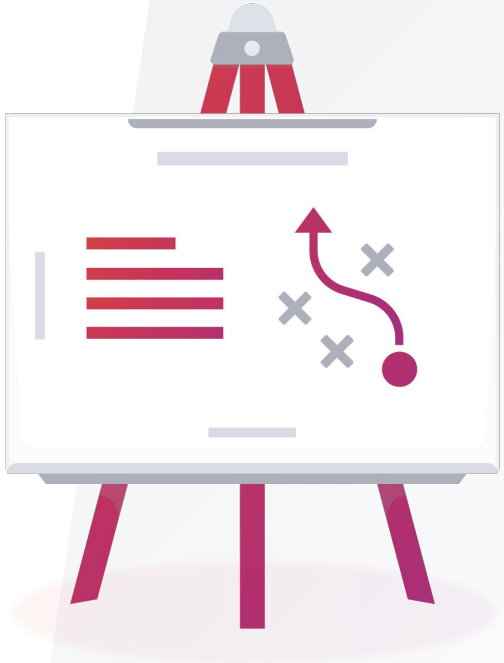
Writing a Beta Test Plan (cont.)

5. **Test Goals**. List the unique goals of the project, such as general quality improvement, interface acceptance, testing real-world performance, testing support infrastructure, collecting customer suggestions, and recording testimonials. Also include the areas (features) of the product that will be tested.

6. **Project Stakeholders**. Define the responsibilities of those involved in the project. Define how feedback will be managed as it changes hands and progresses.

7. **Outline Testing Activities**. Schedule out the activities that you need testers to complete each week in order to evaluate each part of your product. By asking them to use specific features, you'll ensure you're getting feedback about the features that matter most.

8. **Incentives**. Plan out the rewards for participation, as well as the participation levels they'll be based on.



Frequently Asked Question

Q: How long should my beta test last?

A: Beta tests generally run anywhere from **three to six weeks of testing time**. With effective processes and tools, betas can produce great results in as little as two weeks. Beta test length should be determined by the set of goals you intend to accomplish. Depending on those goals, you may also be able to reduce your test length by increasing your tester pool.



centercode



TESTING PROCESS

RECRUITMENT

Tester Recruitment 101

Your tester team is the most valuable resource in your beta test. This team can make or break your entire project. Therefore, finding the right testers is absolutely crucial.

Recruitment is generally split into two phases:

1. Advertise via a variety of channels (e.g., email, social media, etc.) to recruit a large pool of interested candidates. Candidates are generally asked to complete a qualification survey. If possible, it's best for this pool to be 5-10x the size of your intended tester team.

2. Use whatever data you have available to select the most qualified beta testers from the pool of candidates. The more qualified and enthusiastic your selected testers, the fewer total participants you'll actually need in your test.



Why Do People Beta Test?

Knowing why people become testers will help you both recruit great participants and keep them engaged. Some of the primary reasons people want to participate in betas are:

- **They're problem solvers**. Many products (or new revisions of previous products) exist to solve a problem or fill a void of some sort. People feeling the pain of those problems are often eager to be the first to have them solved.
- **They're early adopters**. Some people are enthusiasts who enjoy experiencing early versions of products. These testers love technology and are excited for the opportunity to carry even a minor role in its development process.
- **They're raving fans**. Many people are passionate about a brand or product line and are interested in having a chance to influence where those products go. Beta provides a rare opportunity to do so.
- **They're career-minded**. Business-focused beta tests offer beta testers an opportunity to expand their skill set to include knowledge and use of products that aren't even on the market, gaining them a competitive edge. Similarly, betas can provide an opportunity to network with peers.

Being motivated by earning free products is surprisingly uncommon with good beta testers, and it's generally a motive you'll want to avoid when recruiting.

centercode

Employees, Friends, and Family

The most readily available beta testers are often those closest to you. Unfortunately, they are generally not the best testers.

Employees
Your coworkers are generally a bad choice for real-world tests, as they either have too much insight into the product itself or are biased because of their relationship with the company.

Friends
Friends usually make bad testers because they are inclined to feel that their insider relationship negates the requirement to participate at the same level as other testers.

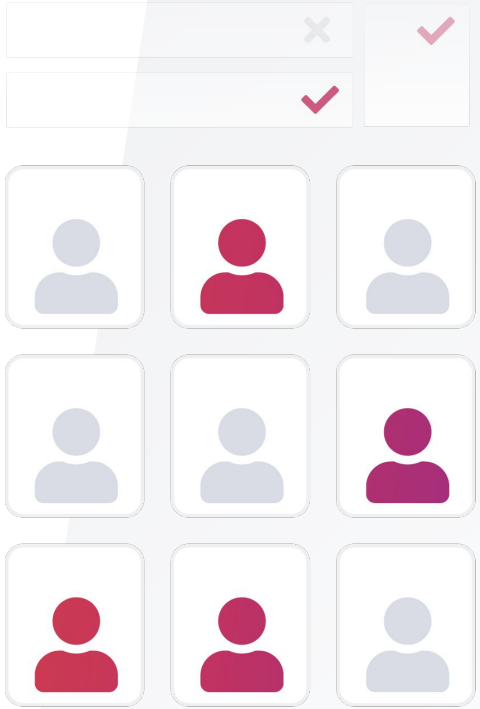
Family
Family members are also not the best choice, as their connection to you or your team may inhibit them from being objective.

Advertising for Testers

- The internet opens the door to an endless supply of beta testers. It's important to manage this resource effectively in order to attract qualified candidates.
- Find portals/networks that interest the target market (every market has them). Ask an administrator to help advertise for you.
- Make the technical requirements for each test very clear. Wading through unqualified candidates is a waste of time.
- If you mention incentives in your call for testers, be vague. Testers are most effective when they're selected based on their enthusiasm about testing rather than promise of reward.
- It isn't necessary to advertise specific features of the product (especially anything confidential). Focus on the product category you're testing.
- Be very clear and honest about the time required to participate in the project.

Selecting Great Testers

- Always keep your target market in mind. Selecting testers from the wrong market is the easiest way to end up with irrelevant data.
- Look for responsive and communicative behaviors. These traits will likely remain constant during the actual beta phase.
- Look for detail-oriented applications. Select candidates whose application responses resemble the quality what you would ultimately like to see in tester feedback during your test.
- Recognize those who are excited about your product.
- Avoid those who refer to or question test incentives immediately.
- If possible, include a few previously successful beta testers. They'll appreciate the recognition and have a positive influence on other testers.



centercode

Frequently Asked Question

Q: How many testers should I recruit?

A: It varies. The size of your tester team depends on your product, target market, prior experience, and the tools available. As a baseline, consumer product tests generally include anywhere from 50-200 people, while business products commonly include only 20-50. Many variables can change these numbers. Keep in mind that the availability of internal resources will also impact the size of your tester team.



centercode



TESTING PROCESS

KICK-OFF

Getting Ready for Beta Testing

centercode

To set your beta test up for success, make sure each of the three main components of your test are ready to go:

1. **Your product is in a test-ready state**. Your product is stable and near feature complete. It should be generally functional, and the majority of testers shouldn't discover unknown issues in their first hour or so of testing.
2. **Testers are both ready and able to test your product**. Testers have been selected, understand the schedule and what's expected of them, and meet any technical, demographic, and knowledge requirements to successfully test.
3. **Your team is ready to engage and react in real-time**. Stakeholders have been identified and educated, and they are ready to interact with testers and feedback.

Missing the mark with any one of these areas will have a negative impact on the quality and quantity of your feedback, ultimately reducing the value and accuracy of your beta test.



Free Resource: Check out the *Getting Ready for Beta Testing* white paper for more on these components.

centercode

Product Readiness Checklist

- ❑ The engineering team has verified that all components of the product are ready for beta testing.
- ❑ Auxiliary components (product documentation, test instructions, etc.) have been assembled into a single package which represents exactly what will be sent to testers.
- ❑ The out of the box experience has been successfully reviewed, including setup, installation, and documentation.
- ❑ Basic product functionality has been successfully reviewed (all key features are working) by product management.
- ❑ Known issues that could not be addressed prior to beta have been clearly documented and communicated to testers.
- ❑ The uninstall process (if applicable) has been successfully verified.

Tester Readiness Checklist

- ❑ A sufficient number of testers for each required target market segment have been selected and notified.
- ❑ Accurate contact information and addresses (if testing hardware) have been verified for all testers.
- ❑ Non-disclosure and beta participation agreements have been explained in plain English, signed, and returned by all testers.
- ❑ Project schedule and tester responsibilities have been clearly communicated to the tester team.
- ❑ Testers understand how to use the systems they'll be using to provide feedback. If they aren't dead simple, training or documentation has been provided.
- ❑ Testers have easy access to all resources needed to carry out their responsibilities and those resources are easy to understand.

centercode

Team Readiness Checklist

- ❑ Core parameters and processes (e.g., testing goals, strategies and mechanisms for collecting feedback, categories for issues, incentives, etc.) have been defined and communicated to all stakeholders and contributors.
- ❑ Milestones and deadlines have been clearly communicated, and all necessary resources are readily available.
- ❑ Stakeholders have delivered all pre-test deliverables (e.g., tools, documentation, surveys, packaging, product keys, NDAs, beta units, etc.)
- ❑ Contingency plans have been defined for any stakeholders with limited availability (e.g., planned vacation, potential birth of a child, etc.)
- ❑ Any infrastructures being used during the beta (test management tools, customer support, issue tracking, content delivery, servers, etc.) are accessible and have been tested.

Legal Agreements

Given the pre-release nature of beta products, most tests require a Non-Disclosure Agreement (NDA) forbidding participants from openly discussing their experiences with the product. Beta NDAs are similar to traditional NDAs, but generally include a few aspects that are unique to beta.

In addition to NDAs, it is common to execute a separate (or combined) agreement known as a Beta Participation Agreement (BPA). This agreement outlines the expectations for participating in the project.

With the introduction of the General Data Protection Regulation (GDPR) in Europe, and other similar laws being passed in the United States, it's also imperative to know where your testers are located and to be cognizant of the personal data you're collecting and how you'll manage it.



Free Resource: We've developed a comprehensive *Beta Test Agreement Kit*, which includes NDA and BPA templates, along with an explanation of their contents.

centercode



TESTING PROCESS

MANAGEMENT

Beta Feedback

As testers experience your product, they naturally tend to provide feedback in the form of issues (bugs to fix), ideas (ways to improve your product), and praise (product aspects to promote). Additional feedback types generated during your test will vary based on the objectives outlined in your beta plan.

Critical feedback types include:

- Issues
- Ideas
- Praise

Other feedback types include:

- Discussions
- Surveys
- Private Journals
- Tester Calls

Issues

Issues are collected in all beta tests. This is generally considered to be the most valuable aspect of beta testing, as correcting defects will lead to higher product quality. The following are tips for collecting beta bugs:

- Issue forms for beta testers are generally simpler than those completed by trained QA staff. At a minimum, they should include a title, description and/or steps to reproduce, and severity. The tester's view of severity may not match your own definition, but it's important to capture their impression of it.
- Issue forms should allow testers to attach files (for screenshots, logs, etc.). These files often describe the issue more clearly or concisely than the tester may be able to (and saves you a great deal of time).
- To reduce the friction of asking testers to provide details about their environment (operating system, hardware configuration, etc.), collect these details upfront in your qualification survey or by profiling the testers in your community.
- Using a comment log for issues makes it easy to communicate back and forth with testers when collecting additional details.
- Testers should be encouraged to report issues not only about the product, but also about the documentation and any other auxiliary systems or services they are experiencing as part of the beta (support systems, etc.). This feedback should be distributed to the appropriate stakeholders.
- Issue forms should flow into your development team's defect tracking system (Jira, BugZilla, etc.). Automating this process saves a great deal of time, allows engineering to use familiar tools, and reduces the opportunity for crucial feedback to be lost.

Ideas and Praise

centercode

Ideas

- Given the near feature complete state of most beta products, the goal of capturing testers' ideas, such as feature requests or other product suggestions, is to gain early insight into customer requirements for future revisions of the product. In this sense, ideas are most useful for product planning.
- If an overwhelming majority of your testers are asking for a specific feature, it may be worth considering that idea for the final version of the current product. This is especially true in business product tests.
- It's a good idea to include a disclaimer on an idea form to claim ownership of any ideas generated by testers as part of the beta test.

Praise

- Most beta testers are more than willing to share their excitement about your product. Praise forms should include instructions to encourage the tester to be open and honest, followed by a freeform text box. If you're looking for a specific length (e.g., for testimonials), it's important to make this clear, as some testers will naturally be very short, while others will be overly verbose.
- Praise is generally collected toward the end of a beta test (possibly as part of a final closure survey) and should include a disclaimer that describes how praise submissions will be used. Depending on your audience, it may also be useful to include a checkbox that confirms the use of the user's name and/or company when referencing it.

Discussions and Surveys

centercode

Discussions

- Discussion forums in beta tests (1) increase participation by providing testers with a sense of community and additional reasons to keep using and discussing the product; (2) reduce beta management and support requirements by allowing testers to support each other; (3) provide a simple mechanism for hosting directed discussions (e.g., an online focus group); and (4) alleviate confidentiality liabilities by providing testers with a controlled outlet to express their excitement, reducing their urge to do so outside of your project (social media, public product forums, etc.).
- While discussion forums are a great tool in beta testing, they're a very inefficient means of managing issues and ideas. It's important to educate testers on the proper channels for submitting different types of feedback.

Surveys

- Surveys are a great way to both collect feedback and keep testers engaged throughout your beta test. These are most commonly used to capture opinions (about features, functionality, or performance) or to collect specific feedback about issues (bugs, etc.) identified throughout the project.
- Generally, surveys should be limited to once per week and include fewer than 20 questions. Open-ended questions require a great deal of effort for both the tester (to write) and the test manager (to read and triage), so it's important to limit the number of questions you include.

Private Journals and Calls

Private Journals

- Journals are simple forms that aren't visible to other test participants. Each one provides a short overview of the tester's recent experience with the product. If possible, these should include a rating scale that allows you to focus on the highly negative or positive experiences.
- Journals are a great way to increase participation, as they give less social users an outlet to discuss their experiences with your product. They also provide a reason to return to your beta site, which will keep testers engaged with updates, builds, etc.

Tester Calls

- Customer conference calls (including either single or multiple testers) offer direct, real-time feedback — similar to a focus group. When used, they are generally scheduled both at the beginning and end of your beta test, offering the product team the ability to converse directly with customers prior to release. Customer calls also increase participation rates by showing testers how valuable their time and feedback are.

Tips for Managing Testers

Achieving high beta tester participation rates is widely considered to be the most frustrating challenge when running a beta test. The following tips will help raise your participation rates.

1. Clearly express expectations early on and throughout
2. Consistently encourage confidentiality (require NDAs, etc.)
3. Balance gratitude with expectations
4. Keep testers involved with ongoing activities
5. Keep testers current regarding project progress
6. Offer a simple method for testers to provide feedback
7. Allow testers to communicate amongst themselves
8. Respond quickly to all issues and requests
9. Contact inactive testers directly (by phone if possible)



Free Resource: Get tips for maintaining high participation in our ebook, the *Tester Engagement Pocket Map*.

Frequently Asked Question

Q: How much time should testers spend testing?

A: This depends on the complexity of the product and the goals of the project. In reality, there is no set time that a tester should dedicate to testing each week. Simple products such as mobile apps might require just 1 - 2 hours a week, made up of the microinteractions users have throughout the day (5 seconds - 2 minutes). Other more complex products like enterprise software might require 5 - 10 hours of testing in a week. Keep in mind that when you're working with properly targeted beta testers, your product will fit into their daily lives and testing shouldn't feel like "work."



centercode



TESTING PROCESS

CLOSURE

Keys to Closing a Beta Test

We're in the home stretch now. The following steps will help you close out your test.

1. **Give your testers time to submit final issues.** Generally, it's best to give your testers at least three days' notice before the end of your project.

2. **Cut-off tester ability to submit feedback.** Once the project reaches its end date, it's important to cut off the flow feedback so you have a clean set of data to report on.

3. **Close all open issues.** Review and close any remaining feedback. This should be done quickly in case additional information is needed before you lose touch with your testers.

4. **Offer testers a simple means to return the product.** If the product is hardware-based, be sure to provide return instructions (testers should never bear any expense).

Incentives

You want to make sure that the testers who provided you with all that amazing and critical feedback feel well rewarded. Providing an incentive will keep those quality testers around for the next time you need their help.

- **Include multiple levels of incentives based on performance**. Participation will likely vary heavily across your tester team. It's important to recognize and thank those who went above and beyond.
- **Different types of incentives are important for keeping different types of testers engaged**. For example, a lunch with a C-level executive might work well for an internal-only dogfooding test. In other cases, a simple thank you email might suffice.
- **Ensure incentives align with your participation requirements**. If you expected a great deal of time out of your participants, the incentive should reflect that level of expected effort.
- **Reward everyone who hits the incentive threshold**. Generally a "top 10" approach to incentives is not as successful as setting thresholds or levels for incentives. If everyone provided incredible feedback, everyone deserves a reward.
- **Distribute incentives quickly**. The golden rule applies here: you expected your testers to participate immediately, so it's only fair for you to do the same. If possible, have your incentives sent within a week of test closure.
- **If possible, thank testers individually**. A personal touch goes a long way.

Developing a Closure Report

Now that your testers are happy and you've sent them on their way, it’s time to get back to your stakeholders. A closure report provides an overview of your beta test results.

- Develop an executive summary of the project
- Document key issues surfaced during testing
- Document frequently requested features
- Document survey results (with pretty charts)
- Highlight your top testers & incentive rewards
- Ensure all departments receive a report

Wrapping Up

8 Elements of Beta Test Success:

1. **A Detailed Plan**. A solid plan provides a clear roadmap for everyone involved in your test.
2. **Universal Buy-In**. A supportive team that believes in beta will be much more willing to help.
3. **Target Market Testers**. The wrong testers produce useless results.
4. **Communication**. Provide regular updates on timelines, responsibilities, and progress.
5. **Responsiveness**. Make testers feel involved and like they are an important part of your team.
6. **Effective Tools**. The right tools increase efficiency across the board.
7. **Organization**. Effectively managing the enormous amount of data generated during beta is key for delivering valuable results.
8. **Tester Acknowledgement**. Testers are volunteers and their efforts deserve to be recognized.

centercode

Delta Testing Solutions

WHY CENTERCODE?

Increase Revenue
½ ☆ INCREASE
in product ratings

Release Sooner
50% FASTER
test cycles

Gain Participation
75% INCREASE
in tester engagement

Save Time
40%+ DECREASE
in time managing tests



The Centercode Platform equips you with the automation and best practices you need to get actionable user insights for any use case.

PRODUCT LAUNCH

Engage your **Target Market** to successfully bring new products and services to market

USE CASES:

- Beta Tests
- User Acceptance Tests
- Field Tests

PRODUCT GROWTH

Recruit **Existing Customers** to continuously mature live products and services

USE CASES:

- Progressive Delivery
- Early Access
- Customer Advisory Boards

CUSTOMER ZERO

Gather **Employee** feedback to enhance your products while fostering a product culture

USE CASES:

- Alpha Tests
- Dogfooding
- Employee Tests

COMPETITIVE EDGE

Enlist **Competitors' Customers** to rapidly gain actionable intel about their products

USE CASES:

- Competitive Intelligence
- Market Testing
- Comparative Research