Gathering feedback from your users regularly is a key to success in any business. The biggest trap you can fall into is to assume that your end user will think and act the same way you do. Therefore, establishing a user testing system through which you will get feedback from your actual customers quickly is crucial for your business to succeed. In this article we will discuss:

1. The importance of usability testing for product teams
2. The concept of continuous user testing and what makes it stand out
3. What to test for, and how it is different from A/B tests

Usability Testing for Development Teams

User testing is a tool that any development team should use, alongside automated tests they run on a regular basis and continuous deployment. The problem is, as with all other tools, user testing takes time to set up, so many teams that engineer products simply ignore this step, thinking that it is way more important to spend time on writing software, rather than running user tests and making adjustments based on the actual feedback they receive.

In the long run, however, factoring in the time spent on creating and updating features that might not get adopted by end users, it becomes apparent that this product development process can be improved at an earlier stage. Of course, the sooner you receive the feedback you need, the easier it is for you to make the correct adjustments, and the greater the impact of the changes you made.

Ongoing user testing is an extremely powerful tool, which can help development teams make numerous iterations and test whether they are going in the right direction with the new changes.

The Advantages of Continuous User Testing

You are probably wondering what this principle of continuous user tests is. At Magora, we use an online testing lab, which allows us to film and quickly obtain videos of our users sharing their experience as they are sampling our product. Once the videos are ready, the whole product team takes the time to view the most current videos.

Essentially, watching these feedback videos has become an integral part of our teams' workflow and is currently one of our highest-leverage activities. This process allows us to react to any part of the user experience that is based on any preconceived notions we have. We deploy several changes and run user tests to get feedback on the updated version. Within an hour or two, we can receive user feedback and watch half-an-hour-long videos to see whether we are moving in the right direction.

This process allows us to learn quickly and brainstorm solutions that can fix any hiccups in the user experience. This process of assertively analysing user tests to substantiate our hypotheses is called continuous user testing.

So what are the advantages of this approach? To understand them better, you should picture the opposite reality, where you only test your product a few times a year or develop something and only test it after it goes into production. By doing this, you are building something that might never take off and there is an extremely high risk that you will need to get rid of what you spent hours, weeks or months building, not to mention the costs of it all. This is especially true if you are building something that you won't use yourself.

Continuous user testing allows you to remain confident in your decisions during every step of the process and at a relatively low cost. It is instrumental in driving the development process of a product, based on your current vision and intuition, which when doing so, can be very easy to get immersed in your own mental model and product vision and get disconnected from what users, who are unfamiliar to the product will think. Gathering continuous feedback helps you adjust your vision to reality. Also, the data you collect assists you in solving internal arguments that can arise within a development team. We often spend hours debating hypothetical concepts where we have no data, only to settle for one vision, which might not be the correct one. With user testing you can save all that energy, test your hypothesis and make sure that what you develop will actually work.

What Should You Test For?

Oftentimes, people get confused between different testing methodologies or stick to a particular one. A/B testing is probably on your mind right now, so you might be wondering why you need the ongoing user testing when you can run split tests and get more measurable results. Well, a good place to start would be to test something that you cannot discover by running an A/B test. In other words, you now have the ability to test something that is not quantifiable and is difficult to measure.

For example, how would you test if you need to know:

- How much do people understand when they take a brief tour of your product?
- How helpful are the tooltips?
- Does a short introductory video lead to users understanding your product better?

- How clear are the requirements to completing certain tasks?
- Why are certain parts of the service not very well understood?

As you can see, the answers to questions like these are very difficult to accomplish with A/B tests. However, when you watch videos, it is much easier for you to understand and see where users get tripped up and why. You can get a much deeper comprehension of the way they behave, which cannot be captured in any numeric metrics.

Conclusion

We encourage you to try the same approach which has helped us succeed. The feedback we continuously get from our users is integral to our development and design processes. We believe that successful software development cannot exist without continuous user testing to get and act on feedback fast and to understand user behaviour much better.

Magora experts can help you to create whatever you are looking for, or answer some questions you may have on developing and testing a product or software.

And if you need some development support, here are links to some of our services:

Need more information about our services? Here are some links:

[Top App Ideas for Startups to Make Money](#)

[Web App Development Cost in 2023](#)

[Generative Design: What is it? How is it being used?](#)

[How to Hire App Developers](#)