

Why Even Small Bugs Lead to Big Problems

Andersen software company presents four reasons why the absence of testing and the resulting small bugs can lead to even loss of business.

NEW YORK, NEW YORK, USA, August 25, 2020 /EINPresswire.com/ -- Nowadays, the end-user is no longer amused that only the key functions of a product work properly - the visual component and the flawlessly working mechanism are important. Minor defects in the UI can encourage the user to leave negative comments about the product.

[Andersen](#) has been providing software testing services for over 13 years. They say their clients are often trying to save on testing. Every time specialists have to explain why testing is needed and how its absence or insufficient quality can affect the final result.

Some customers, in pursuit of a quick product release, are reluctant to spend time and money on testing and fixing minor defects or put this task on the back burner due to its low priority. They're usually sure that, first of all, it's necessary to launch the project, make money, and only then set up the process of testing and debugging the application. However, one shouldn't be tempted to save on testing at the initial stages, because this decision can lead to significant problems.

1. Criticism in app stores and on social media

Criticism of a product is the first and foremost consequence of ignoring minor bugs. When something annoys users, they write about it in reviews and discuss it on social networks. If there are thousands of such users, a wave of negativity rises, which can cost a business millions.

Applications with errors, even non-critical ones, look unprofessional, negligent, and turn off customers.

2. Small errors invite the assumption of big errors

Let's consider a minor non-critical bug, for example, a misspelling in a pop-up message of a Healthcare application. When a doctor adds a medicine to a prescription, a confirmation message pops up asking "Do you want to add Parrocetamol?"

It may seem funny at first, but over time, it becomes annoying. And the most unpleasant thing is that the healthcare specialist may develop distrust in the application. How can the app be

trusted with patients' health and lives if the team can't fix such a simple bug as a spelling mistake? Careless embarrassing mistakes cost companies the trust and respect of their customers.

3. Errors in UI hinder performance

When developing an application interface, it's necessary to proceed from the needs and convenience of the user. When dealing with a user-hostile application, people have to perform unnecessary actions, and this reduces productivity.

There are several ways of improving UI in order to prevent underperformance and outflow of users:

- * Conducting comprehensive testing before release and releasing the application only after all bugs have been eliminated.
- * Studying the experiences of various user categories in order to gain an understanding of various performance parameters.
- * Providing the user with the ability to independently customize the application the way they want.

4. Small errors in calculations entail large losses

A bug in a FinTech application, causing a transaction to be carried out with an error of a few cents, can lead to the loss of hundreds of thousands of dollars per year if the client is actively involved in financial activities. Still, at the testing stage, the person in charge might ignore the signal from QA engineers that 1 cent was lost during a transaction of \$1,000 - after all, this is only 0.001%.

Glitches in applications for calculating medicine doses or load-bearing capacity of constructions can be even more dangerous. Of course, the error won't go unnoticed at the stage of the specialist's work. But this means that the doctor or engineer will have to check all the calculations manually, which leads to wasted time, decreased productivity, and irritability.

Conclusion

The prevailing opinion is that it's important to fix significant bugs before release, and small ones can be left for users to catch. But users may be not over-enthusiastic with that.

Cutting corners on finding and fixing bugs can lead to much more serious losses. If the release of the application is not strictly scheduled, postponing it for several days or weeks is not a bad decision. If postponing the release is not possible, one can always involve additional resources. This opportunity is provided by outsourced testing services as those Andersen provides.

Vladimir Nichiporovich

Andersen

+1 917 993 9742

[email us here](#)

This press release can be viewed online at: <https://www.einpresswire.com/article/524774524>

EIN Presswire's priority is source transparency. We do not allow opaque clients, and our editors try to be careful about weeding out false and misleading content. As a user, if you see something we have missed, please do bring it to our attention. Your help is welcome. EIN Presswire, Everyone's Internet News Presswire™, tries to define some of the boundaries that are reasonable in today's world. Please see our Editorial Guidelines for more information.

© 1995-2020 IPD Group, Inc. All Right Reserved.