



## #72 Consensus bias

Because we view mistakes from a business perspective, we sometimes exaggerate their significance to users. Therefore, before we decide what to do, we need to understand the real significance of the mistake through the user's eyes. Often, the issue can be fixed without further mentioning it because of its insignificance for users.

## #68 Pro-innovation bias

Sometimes we exaggerate the significance of mistakes because of our very personal approach to business. Therefore, we should double-check our assumptions with the team.

## #101 Peak-end rule

We need to understand what emotions this mistake causes. To do this, we need to analyze the workflow in which the error occurs and determine its "painfulness." The intensity of negative feelings caused by mistakes will show us the form of communication that we will choose.

## #83 Loss aversion

The most critical errors are those related to data loss. In addition to an early apology, we should explain in detail to users that the situation is under control and show the steps that we will take to complete data recovery. If the losses are irrecoverable, then depending on the specifics of the product, we are obliged to offer some free bonuses and other preferences.

## #3 Illusory truth effect

Suppose we know that the mistake will be discussed for a while. In that case, we should develop a clear position, formulate it and repeat it systematically until people use it to rationalize what happened. This technique is widely used by politicians.



## #7 Mood-congruent memory bias

We can report an error on a day that is colored with positive associations. In doing so, we can mix the message with some good news. For example, in the next release, we can tell our users about six innovations, five fixed bugs, and one bug found in the last release. Simultaneously, if the error is very painful for users, we can write out a detailed plan to fix it so that users have a vision that we have everything under control.

## #10 Omission bias

We should check our ideas with teammates. Sometimes, doing nothing and ignoring the mistake can be the solution (many companies do this). However, in some cases, ignoring it can cause irreparable damage.

## #22 Framing effect

When publicly acknowledging the error, we should avoid wording in which users can "see themselves" ([#16 Self-reference effect](#)). Also, we should avoid using sentences about possible terrible consequences of our mistakes ([#11 Base rate fallacy](#)).

## #51 Placebo

Sometimes it can be helpful to give users some kind of button to provide them with emotional comfort. For example, you can add a button "Report a bug" on a page where we already know a bug. Such a move can be useful if we need a little time (24 hours?) to fix the error.

## #66 Hindsight bias

Depending on the public discussion that began after the publication of the mistake, there will certainly be "experts" who will say that "it was obvious," "only a matter of time," etc. We should not react to such attacks because it is useless. At the same time, we should suppress the same sentiments within our team.

## #100 Fading affect bias

Even big mistakes can be healed over time. We should never stop product development. Updates should be released normally. We should avoid situations where we inadvertently increase the significance of the problem so that the memory of it fades out much later ([#17 Negativity bias](#)).