3di Software Usability

10 tips for software usability

Clients sometimes ask us to take a quick look at their software and tell them what the main usability issues are and how to fix them. This is trickier than you might think. But we've done a lot of usability testing, so we do know the common places that usability issues occur.

Based on our experience, combined with insight from our network of user experience professionals, here is our list of 10 things you can do to start making improvements to your software's usability.

1 Sort out your terminology

Terminology needs to be clear, unambiguous, consistent and match the way users understand what they're trying to do. A good starting place: if your configuration options or dialogue boxes have names similar to functions or classes in the code, then your UI is using developer terminology – which users are probably not going to understand.

2 Check your consistency

If several developers have worked on different areas of the UI, it's likely that similar functions are named and work in slightly different ways. For example, *Save changes* in a configuration dialogue works differently from *Save changes* in a user management area.

This confuses users: without realising it, they learn how things behave… and are slowed down by things behaving differently.

3 Rewrite your error messages

One project we worked on reduced support calls by 15% by doing this step alone. Error messages should tell people what's caused the error and what they should do about it. If users need more help understanding the error, they contact your support team or they search online – so make sure the message also includes an error number or short title to help them with this.

4 Reduce the number of errors that users encounter

Users don't like errors. It's obvious … and confirmed in tests by responses of panic/terror/despondency whenever an error or warning pops up.

In particular, look for errors that are generated because of user behaviour... and find a way to prevent or catch that behaviour earlier. For example, could you validate individual fields on a form before allowing the user to click *Submit*; give guidance on acceptable values for inputs; check for system requirements on installation or start-up..

5 Remove a feature

Some research by the Standish group nearly a decade ago found that over 60% of software features were rarely used or never used.

Superfluous features get in the way of your users finding the good stuff. So if you've got features that have crept in during development "just in case the user wants to do x" … consider taking them out and making sure they really are of value to users before you add them back in.

www.3di-usability.com
01483 211533

6 Rename buttons

Naming buttons helpfully speeds users up and reduces the number of mistakes they make. To illustrate: a dialogue box asks users: "You have unsaved changes in your file. Would you like to save your changes before you close the file?"

Which button is easier to understand? Yes Save changes

7 Put some "help" in the UI

Through our own research, we know that users do make use of tooltips, pop-ups and links to useful information or video demos from the UI.

For options that users struggle to choose between, add some helpful text … or for a button that users aren't sure about clicking, add a tooltip that explains what will happen if they click … or for a screen where users have to take some action to get started, use the blank space on the screen to present some getting-started tips or link to your video demo.

We don't recommend putting these widgets everywhere in your UI, "just in case", though. In fact, that's counter-productive: users quickly learn not to trust tooltips if they come across ones that just repeat the text they can already see in the UI. So use with care!

8 Check your defaults

Could empty fields be populated with data you have about the user or their system? Are the defaults the most likely values that users will suit users? Some research by a US-based user experience consultancy found that 95% of users don't change from default configurations settings, so you need to get this right..

9 Only offer functionality that really is available to the user

Double-check that you've disabled options, buttons or fields that don't apply … and only re-enable them when they do apply. It's usually best not to hide this functionality completely, though – that will confuse people when it appears later.

10 Check your end-to-end workflows for key tasks

We've saved the most common cause of usability issues until last…

Your software is almost certainly a combination of pieces of functionality created by different people, over a long period of time. Features are implemented well, but don't quite fit together in the workflow - e.g. do you ask the user to re-enter data they've already entered, or expect the user to have completed a step that they're unlikely to have done yet? Or maybe the user has to open multiple dialogues to make complete a task?

Step through each of the key tasks end-to-end (with default settings!) to identify things that need to be sorted out.

3di Software Usability

Contact us if you'd like to discuss how we can help you understand your users, identify and prioritise usability issues, and design solutions.

www.3di-usability.com
01483 211533