Developer Forums

Post

Bug Reporting: How and Why?

⚠ This thread has been locked by a moderator.

I file a lot of bug reports. I also ask a lot of third-party developers to file bug reports about problems they see. This post is my attempt at collecting together some hints and tips about Apple's bug reporting process.

To file a bug report, use Feedback Assistant.

To file an enhancement request, use Feedback Assistant. In the "What type of issue are you reporting?" popup, choose Suggestion.

Feedback Assistant has another popup labelled "Which area are you seeing an issue with?" This has two sections, with user-level components at the top and developer-level components at the bottom. If you're filing a bug against an API, choose a developer-level component. For example, if you're filing a bug against Security framework, scroll past the Security component in the user-level section to the Security Framework component in the developer-level one.

Feedback Assistant is both a website and an app. For details on how to access the app, see *Installing the App* on the Feedback Assistant for Developers page.

For lots of great hints and tips about filing bugs, see File effective bug reports. I may repeat some of the info covered by that article, but only where I think it's particularly important.

After filing a bug report, please include the bug number in your DevForums post. Feedback Assistant shows the bug number in both the bug list and the bug detail view. Bug numbers start with `FB` and continue with a sequence of digits. For example, a recent bug report posted here on DevForums was `FB11357055`.

Including the `FB` prefix avoids any confusion as to what type of bug number this is. Occasionally you'll see other types of bug numbers. In release notes they are just in parens, so `(99071807)`, and in DTS documents they have an `r.` prefix, so `(r. 99071807)`. These are Radar numbers. Radar is the internal bug system here at Apple.

Other developers can't see your bug report. However, posting the bug number is still important for a couple of reasons:

- It allows Apple folks to quickly track down your issue.
- If another developer wants to file a related bug, they can reference your bug number in their bug report.

You *can* share bug reports with other members of your team. For details on how to do that, see *Sharing Feedback with Your Team* on the Feedback Assistant for Developers page.

Otherwise you can't see a bug report filed by another developer. One option here is to file your own bug and request that it be marked as a duplicate of the original bug. You'll then be notified when the original bug is closed.

The File effective bug reports article discusses the importance of including a sysdiagnose log with your bug report. I have a few additional tips:

- Include a sysdiagnose log from the machine that's having the problem. Sometimes I see folks with an iOS problem include a sysdiagnose log for their development Mac)-:
- Include a sysdiagnose log even if you attach a test project that reproduces the problem. The contents of the log help with screening, making sure that your bug gets to the right person quickly. And that person will be very grateful to have your test project!
- If multiple machines are involved — for example, you're reporting a Handoff problem — include a sysdiagnose log from each machine.
- Trigger the sysdiagnose log as soon as possible after you see the problem. The sooner you trigger the log, the more likely it is to capture relevant info.
- If you install a debug profile, install it *before* you reproduce the problem. These profiles typically enable more logging, and you want that logging to be active at the time of the problem.
- Most profiles activate immediately but some require that you restart. Follow the instructions that came with the profile.
- When reproducing the problem, make a rough note of the time and include that in your bug report. This doesn't have to be super precise. Just note down the time, with seconds, shortly after you reproduce the problem.
- If the bug involves a crash, reproduce the crash outside of Xcode and then take your sysdiagnose log. This means running your code as the user would, so running an app from the Finder on macOS, running an app from the Home screen on iOS, and so on. If you reproduce the crash inside Xcode, the Xcode debugger catches the crash and prevents the system from generating a crash report to include in the sysdiagnose log.

Finally, some personal notes:

- DevForums is not an official support channel. Mentioning a problem here is not the same as filing a bug about it. That's true even if you're discussing your issue with an Apple person. Apple is a big company, and it's very unlikely that the person you're talking to is directly responsible for fixing your bug.

[There'll be more personal notes to come in the future I'm sure (-:]

Share and Enjoy
—
Quinn "The Eskimo!" @ Developer Technical Support @ Apple
`let myEmail = "eskimo" + "1" + "@" + "apple.com"`

Revision History

- **2022-09-01** Added a personal notes section at the end. Added a new bullet in the sysdiagnose section about running your app outside of Xcode. Made other minor editorial changes.
- **2022-08-29** Added a note about filing duplicate bugs.
- **2022-08-24** First posted.

Forums Feedback   Feedback Assistant

Reply

Posted 1 month ago by   eskimo

| Add a Comment

 Developer ⟩ Forums

Platforms

iOS
iPadOS
macOS
tvOS
watchOS

Tools

Swift
SwiftUI
SF Symbols
Swift Playgrounds
TestFlight
Xcode
Xcode Cloud

Topics & Technologies

Accessibility
Accessories
App Extensions
App Store
Audio & Video
Augmented Reality
Business
Design
Distribution
Education
Fonts
Games
Health & Fitness
In-App Purchase
Localization
Maps & Location
Machine Learning
Security
Safari & Web

Resources

Documentation
Curriculum
Downloads
Forums
Videos

Support

Support Articles
Contact Us
Bug Reporting
System Status

Account

Apple Developer
App Store Connect
Certificates, IDs, & Profiles
Feedback Assistant

Programs

Apple Developer Program
Apple Developer Enterprise Program
App Store Small Business Program
MFi Program
News Partner Program
Video Partner Program
Security Bounty Program
Security Research Device Program

Events

App Accelerators
App Store Awards
Apple Design Awards
Apple Developer Academies
Entrepreneur Camp
Tech Talks
WWDC