

# How to Get End-User (Client) Feedback on Custom Development Projects

Asked 16 years, 2 months ago    Modified 9 years, 3 months ago

Viewed 997 times

---



6



My company is a custom development shop for a number of projects, some larger and some smaller. Currently we handle all of our client communication through email. So we email a design doc, they mark it up and send it back. Then we roll out a beta version of their product and they email us with any bugs, new features, etc. And so on....



As I am working on implementing a new bug tracking system (it looks like it will be Mantis right now), I got to wondering how we could best allow our customers an interface with our development process that would provide better tracking of feature requests and client submitted bugs as well as communicate our responses back to the client.

If anyone is aware of a a bug tracking system that does this exceptionally well I'd be interested to hear of that. Otherwise I'm just looking for some general guidelines or good business practices that have allowed your companies to interface effectively and efficiently with your clients.

**UPDATE:** My company uses a LAMPP stack and as we are a small shop with a limited budget we tend to stick to tools that are open-source and free.

Do most people either use Team Foundation Server to handle this or emails back and forth?

bug-tracking

user-feedback

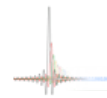
end-user

Share

Improve this question

Follow

edited Aug 31, 2015 at 15:26



Marcus Müller

36.2k ● 4 ● 57 ● 100

asked Oct 23, 2008 at 16:07



Noah Goodrich

25.2k ● 14 ● 68 ● 97

4 Answers

Sorted by:

Highest score (default)



3



I think the key is to have the dedicated tracking system there for bugs/requests, and to establish a set process for communication. With that at minimum you will start getting consistent feedback. From there you can tweak it to get your specific needs.



As an aside, rather than just using e-mail for your communication, I strongly recommend going to smething like [BaseCamp](#) for a project management tool. I find that

it helps greatly with keeping messages, documentation, and timelines communicated to the client.

Share Improve this answer

answered Oct 23, 2008 at 18:11

Follow



**Mitchel Sellers**

63.1k ● 15 ● 114 ● 174



2



If you are using Team Foundation Server, I recommend you to install [TeamPlain Web Access](#). They allow you to expose a web interface to your TFS project. The only things left to do, is give rights to your client and a username and a password.



Otherwise, there is some paying tools like [FogBugz](#). Of course, the principal is having to bug reporting tools directly linked to your Source Control so that the developers can easily fix bugs.

Share Improve this answer

answered Oct 23, 2008 at 16:13

Follow



**Maxime Rouiller**

13.7k ● 9 ● 60 ● 108

TeamPlain is indeed a great product. Without it I consider TFS only half the problem (source control and automated build). – [cfeduke](#) Oct 23, 2008 at 17:10



1

Although I know of no specific tools (at least no open source ones), I suggest that you setup a system which will cover your overall requirements gathering and implementation process. Requirements could be tracked



in the system, which would also contain the design documents (which could be "checked out from" and "committed to" the system). This way, you would tackle the problem of having multiple revisions of design documents around. Additionally, the design documents and the requirements could be tracked easily. If this system were linked to your source code management system, you would additionally ease your development process/requirements tracking.

Share Improve this answer

answered Oct 23, 2008 at 19:02

Follow



MP24

3,200 ● 22 ● 23



Another possibility is to use two products in concert, here's our current setup with a team of 12:

1

[osTicket](#) for incoming requests from clients



- Allows for issues to be handled by support staff and bugs to be verified
- Status can be checked with just an email address and ticket ID
- Typically users don't submit detailed enough bug reports so is a good first step

[redmine](#) for development tickets

- Ticket created by QA or a developer if issue is a real bug

- Provides solid enough project and release management
- Is a solid step up from [trac](#) and [mantis](#) (and provides [migration](#) tools)

Share Improve this answer

edited Feb 16, 2009 at 20:23

Follow

answered Feb 16, 2009 at 15:54



Apex

1,018 ● 1 ● 11 ● 13

---