CS-401

Professor Wurst

3-3-2012

|  |  |  |
| --- | --- | --- |
| **SIZE:** |  |  |
| The source code is more than 100 MB. [ +5 points of FAIL ] |  |  |
| If the source code also exceeds 100 MB when it is compressed [ +5 points of FAIL ] |  |  |
|  |  |  |
| **SOURCE CONTROL:** |  |  |
| There is no publicly available source control (e.g. cvs, svn, bzr, git) [ +10 points of FAIL ] |  |  |
| *There is publicly available source control, but:* |  |  |
| There is no web viewer for it [ +5 points of FAIL ] |  |  |
| There is no documentation on how to use it for new users [ +5 points of FAIL ] | +3 | The Wiki contains an API, but the explanation on how to use it is too technical. |
| You've written your own source control for this code [ +30 points of FAIL ] |  |  |
| You don't actually use the existing source control [ +50 points of FAIL ] |  |  |
|  |  |  |
| **BUILDING FROM SOURCE:** |  |  |
| There is no documentation on how to build from source [ +20 points of FAIL ] | +10 | The documentation exists, but it is a blog post that is not easily found. |
| Documentation exists on how to build from source, but it doesn't work [ +10 points of FAIL ] |  |  |
| *The source is configured:* |  |  |
| with a handwritten shell script [ +10 points of FAIL ] |  |  |
| by editing flat text config files [ +20 points of FAIL] |  |  |
| by editing code header files manually [ +30 points of FAIL ] |  |  |
| The source isn't configurable [ +50 points of FAIL ] |  |  |
| *The source builds:* |  |  |
| using something that isn't GNU Make [ +10 points of FAIL ] |  |  |
| only with third-party proprietary build tools [ +50 points of FAIL ] |  |  |
| with your own build tool for this code [ +100 points of FAIL ] |  |  |
| The build has no tests [ +5 points of FAIL ] |  |  |
|  |  |  |
| **BUNDLING:** |  |  |
| Your source only comes with other code projects that it depends on [ +20 points of FAIL ] |  |  |
| If your source code cannot be built without first building the bundled code bits [ +10 points of FAIL ] |  |  |
| If you have modified those other bundled code bits [ +40 points of FAIL ] |  |  |
|  |  |  |
| **LIBRARIES:** |  |  |
| Your code only builds static libraries [ +20 points of FAIL ] |  |  |
| Your code can build shared libraries, but only un-versioned ones [ +20 points of FAIL ] |  |  |
| Your source does not try to use system libraries if present [ +20 points of FAIL ] |  |  |
|  |  |  |
| **SYSTEM INSTALL:** |  |  |
| Your code tries to install into /opt or /usr/local [ +10 points of FAIL ] |  |  |
| Your code has no "make install" [ +20 points of FAIL ] |  |  |
| Your code doesn't work outside of the source directory [ +30 points of FAIL ] |  |  |
|  |  |  |
| **CODE ODDITIES:** |  |  |
| Your code uses Windows line breaks ("DOS format" files) [ +5 points of FAIL ] |  |  |
| Your code depends on specific compiler feature functionality [ +20 points of FAIL ] |  |  |
| Your code depends on specific compiler bugs [+50 points of FAIL] |  |  |
| Your code depends on Microsoft Visual Anything [ +100 points of FAIL ] |  |  |
|  |  |  |
| **COMMUNICATION:** |  |  |
| Your project does not announce releases on a mailing list [ +5 points of FAIL ] | +3 | Although I haven't seen the Eutester pop up in the Eucalyptus mailing list, it may have. |
| Your project does not have a mailing list [ +10 points of FAIL ] | +10 | The project is too small at this point. |
| Your project does not have a bug tracker [ +20 points of FAIL ] |  |  |
| Your project does not have a website [ +50 points of FAIL] |  |  |
| Your project is sourceforge vaporware [ +100 points of FAIL ] |  |  |
|  |  |  |
| **RELEASES:** |  |  |
| Your project does not do sanely versioned releases (Major, Minor) [ +10 points of FAIL ] | +10 | The project is only available as source code from github. |
| Your project does not do versioned releases [ +20 points of FAIL ] |  |  |
| Your project does not do releases [ +50 points of FAIL ] |  |  |
| Your project only does releases as attachments in web forum posts [ +100 points of FAIL ] |  |  |
| Your releases are only in .zip format [ +5 points of FAIL ] |  |  |
| Your releases are only in OSX .zip format [ +10 points of FAIL ] |  |  |
| Your releases are only in .rar format [ +20 points of FAIL ] |  |  |
| Your releases are only in .arj format [ +50 points of FAIL ] |  |  |
| Your releases are only in an encapsulation format that you invented. [ +100 points of FAIL ] |  |  |
| Your release does not unpack into a versioned top-level directory (e.g. glibc-2.4.2/ ) [ +10 points of FAIL ] |  |  |
| Your release does not unpack into a top-level directory (e.g. glibc/ ) [ +25 points of FAIL ] |  |  |
| Your release unpacks into an absurd number of directories (e.g. home/johndoe/glibc-svn/tarball/glibc/src/) [ +50 points of FAIL ] |  |  |
|  |  |  |
| **HISTORY:** |  |  |
| Your code is a fork of another project [ +10 points of FAIL ] |  |  |
| Your primary developers were not involved with the parent project [ +50 points of FAIL ] |  |  |
| *Until open sourcing it, your code was proprietary for:* |  |  |
| 1-2 years [ +10 points of FAIL ]  3-5 years [ +20 points of FAIL ]  6-10 years [ +30 points of FAIL ]  10+ years [ +50 points of FAIL ] |  |  |
|  |  |  |
| **LICENSING:** |  |  |
| Your code does not have per-file licensing [ +10 points of FAIL ] |  |  |
| Your code contains inherent license incompatibilities [ +20 points of FAIL ] |  |  |
| Your code does not have any notice of licensing intent [ +30 points of FAIL ] |  |  |
| Your code doesn't include a copy of the license text [ +50 points of FAIL ] |  |  |
| Your code doesn't have a license [ +100 points of FAIL ] |  |  |
|  |  |  |
| **DOCUMENTATION:** |  |  |
| Your code doesn't have a changelog [+10 points of FAIL] |  |  |
| Your code doesn't have any documentation [ +20 points of FAIL ] |  |  |
| Your website has no examples of use [ +20 points of FAIL ] |  |  |
| Your website doesn't have any documentation [ +30 points of FAIL ] |  |  |
|  |  |  |
| **TOTAL FAIL:** | +36 | Most of this fail comes in the form of limited documentation and low-visibility. |

I realized after the first pass that the assignment was to assess the Eucalyptus project not the Eutester. But, I found very little wrong with Eucalyptus, and I believe assessing the Eutester will be more beneficial. Because the tester is such a small side-project to Eucalyptus there is an apparent lack of visibility on the project itself. The software is not easily found from the Eucalyptus website, and there is little to no documentation on how to use it properly. According to my assessment 'Babies cry when your code is downloaded'. This may be a bit harsh for a project in its infancy, but clearly there is work to be done.