He estado documentandome sobre Qt y alternativas debido a que por lo que he visto [Qt Designer puede funcionar para cualquier lenguaje de programación soportado por Qt](http://en.wikipedia.org/wiki/Qt_%28framework%29#Bindings). Y... si no podemos aportar nada nuevo para que reinventar la rueda.

Además Qt tiene restricciones en su licencia de uso, de los cuales no me he informado pero sé que por hacer determinadas cosas tienes que pagar. Como por ejemplo compilar las librerias de Qt dentro de tu mismo programa para crear solo un archivo binario sin dependencias.

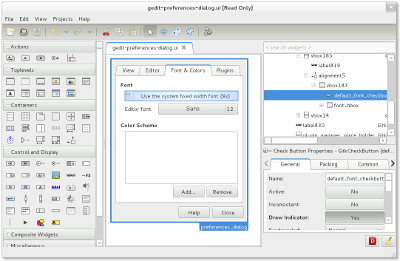
Sobre wxWidgets había mucha opinión variada, pero he pasado directamente a la [página](http://www.gtk.org) de **GTK+**.

**Licencia**: licensing terms for GTK+, the [GNU LGPL](http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html), allow it to be used by all developers, including those developing proprietary software, without any license fees or royalties.

**Características**: “superb performance”, [language bindings](http://www.gtk.org/language-bindings.php), cross-platform, **Theme support**, **GtkBuilder** (creates your user interface from XML) y mobile noseque... parecen widgets para móvil como jquery mobile pero creo que solo son compatibles con determinados móviles.

Ejemplos de temas visuales (Themes): [Uno](http://0rax0.deviantart.com/art/Hope-GTK-187658025), [dos](http://fc07.deviantart.net/fs71/i/2012/074/6/1/greyness_green_gtk_theme_v1_by_craazyt-d4sodn4.png), [tres](http://fc05.deviantart.net/fs71/i/2010/364/5/3/deviantart_gtk_by_therealpadster-d30tf7v.png), [cuatro](http://fc02.deviantart.net/fs37/i/2008/251/c/8/DewDrop_GTK_Emerald_Updated_by_chh_gx7.png), etc.

GTK+ También dispone de una RAD para diseñar interfaces llamada [Glade](http://glade.gnome.org/):



Open source, pero solo permite diseñar los distintos tipos de ventanas y exportarlas a XML, luego cargarlas a tu programa mediante código llamando al GtkBuilder que construye una interfaz mediante XML. Esta aplicación nos permite añadir nuevos widgets e [integrar la “Glade UI Designer core library” en nuestra aplicación](http://developer.gnome.org/gladeui/3.6/) como ha hecho el IDE [adjunta](http://projects.gnome.org/anjuta/), que ha integrado Glade dentro de su aplicación. (INFORMACIÓN INCORRECTA)

Glade no dispone de proyectos ni edición de código, sólo crear nuevas ventanas y obtener su XML. Tampoco dispone de templates.

Propongo tres posibles opciones si acabamos escogiendo GTK+ en vez de Qt::

1. Programar nuestro propio Glade, aka reinventar la rueda (pues vaya)
2. Hacer un fork del código fuente para añadir las opciones de proyectos y templates, de forma que se puedan crear proyectos para un lenguaje específico y ya te cree el código base pudiendo elegir un “Theme” específico o importar otro.
3. Crear nuestra aplicación que permita edición de código, creación de proyectos, etc e integrar el “Glade UI Designer core library” para la edición de GUIs. (INCORRECTO)

Como editor de código existe un componente llamado [Scintilla](http://www.scintilla.org/), compatible con GTK+ que ya nos hace toda la faena de syntax highlighting, code completion, indicación de errores y esas cosas.

Por simplicidad, yo me quedaría con la segunda opción, aunque conlleva su faena de entender la estructura del código fuente de Glade.

EDIT: [Tabla de lenguajes](http://en.wikipedia.org/wiki/GTK%2B#Programming_language_bindings) por los que se han hecho bindings de GTK+.

En la mayoría de lenguajes **conocidos** se puede trabajar con GTK+ en cualquier plataforma, excepto el binding de java, que no funciona en Windows.