

Utilización y administración avanzadas de sistemas GNU/Linux y aplicaciones Software Libre para estudiantes universitarios

Recursos de programación en GNU/Linux

Ramón Rey Vicente rrey@ramonrey.net

Recursos de programación en GNU/Linux

- Lenguajes
- Frameworks
- Entornos de desarrollo
- Otros recursos
- Colaborando



Lenguajes

- C
 - C++
 - Java / C#
 - Python / Perl / Ruby
 - Otros: Objective-C, Fortran, Eiffel, Haskell
-
-

C

- La base de GNU/Linux y UNIX
- Gran cantidad de bibliotecas
- glibc, GCC



C++

- Permite usar muchos de los recursos disponibles en C
- Problemas de estandarización de compiladores
- Ventajas en el diseño e implementación de aplicaciones
- GCC



Java

- Ya es libre !!!
 - Existen implementaciones ajenas al control de SUN
 - Independiente de plataforma, en teoría
 - El mejor camino para migrar de entornos de desarrollo privativos a entornos libres
 - JDK, GCJ
-
-

C#

- “Java” de Microsoft
 - MONO
 - Buen framework de desarrollo, moderno
 - Intenta seguir a Microsoft con su .NET
 - Incompleto, siempre por detrás
 - Complicado migrar de MONO a .NET y viceversa
-
-

Python

- Multiplataforma
- Dinámico
- Orientado a objetos
- Sencillo
- Web y escritorio



Perl

- Multiplataforma
- Hereda features de C, sh, Awk
- Potente manejo de XML
- Fácilmente trabaja con C/C++
- DBI, simplifica acceso a BBDD



Ruby

- Multiplataforma
- Dinámico
- Javascript + Python + Java
- Fácil extender en C



Otros

- Haskell
 - Eiffel
 - Fortran
 - R
 - ML
 - Objective-C
-
-

Frameworks

- Java
- Mono / .NET
- QT / KDE
- Gtk+ / GNOME



Java

- Bytecode
 - Recolector de basura
 - JDBC, acceso a bases de datos
 - aplicaciones web, servlets
 - introspección, plantillas
 - manejo de xml, cifrado y seguridad
 - programación distribuida
-
-

Java (II)

- Swing/AWT para aplicaciones gráficas
- Soporte de Ruby además de Java
- ... hasta tiempo real



Mono / .Net

- clon de Java
 - integra XML, multihilo, seguridad y cifrado
 - interfaz gráfica con gtk#/gnome#
 - soporta Python, Perl, Javascript, VB.net, Java, etc además de C#
 - no es recomendable para multiplataforma
-
-

QT / KDE

- creación de aplicaciones gráficas QT/KDE
 - manejo de multihilo
 - XML
 - red
 - tipos básicos portables
 - KDE: multimedia, manejo de archivos, componentes con dcop
-
-

gtk+ / GNOME

- creación de aplicaciones gtk/GNOME
- tipos básicos portables (glib)
- manejo de gráficos, animaciones
- GNOME: manejo de archivos, multimedia (gstreamer),



Entornos de desarrollo

- Eclipse
 - Netbeans
 - Anjuta
 - Kdevelop
 - Vim / Emacs
-
-

Eclipse

- multiplataforma
 - toolkit propio SWT, nativo para cada plataforma
 - soporta Java, C++, PHP, a través de plugins
 - soporta múltiples SCM
 - editores visuales (incompletos)
-
-

Eclipse (II)

The screenshot displays the Eclipse IDE interface. The Package Explorer on the left shows a project structure with a package named `org.jaxup.samples.jdom` containing the file `JDomUpdateDemo.java`. The main editor window shows the source code of `JDomUpdateDemo.java`. The code includes a `main` method that checks the number of command-line arguments and uses `SAXBuilder` and `XUpdate` to update an XML document. A tooltip is visible over the `updater.updater` line, providing details about the `XUpdate` class and its `runUpdate` method. The tooltip text is as follows:

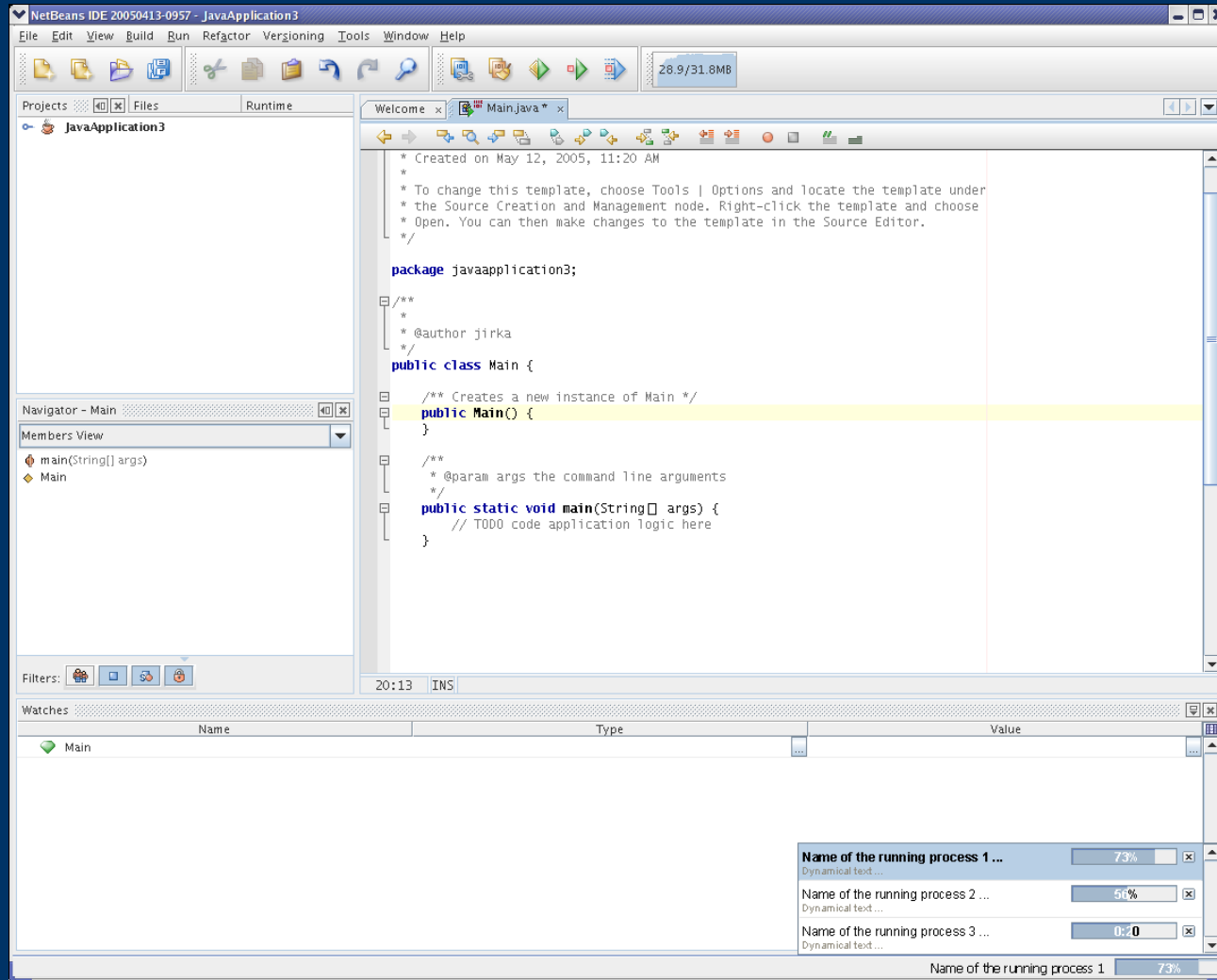
```
Runs a series of xupdate instructions to manipulate a given source document.  
Parameters:  
sourceDoc the source document that is going to be changed.  
updateContainer an element node that contains xupdate instructions as its children. Ordinarily, this would be the document element of an xupdate XML document.
```

Below the tooltip, a red error message is displayed in the editor's status bar: `updater.updater cannot be resolved or is not a field JDomUpdc: jaxup/src/java/samples/org/jaxup/samples/jd line 42`. The IDE's status bar at the bottom indicates the current cursor position as `Writable Insert 42 : 21`.

Netbeans

- Java, C++ y Ruby
 - Editor visual de aplicaciones Java
 - Muy extensible: plugins
 - Multiplataforma
 - Requiere bastantes recursos
-
-

Netbeans (II)



Anjuta

- Orientado a gtk+/GNOME
 - Integración con autotools (automake, autoconf, etc)
 - Integración con gdb
 - Soporte para Pascal, Perl, Java, por archivo
 - Ligerero, ideal para máquinas modestas
-
-

Anjuta (II)

The screenshot displays the Anjuta IDE interface. The main window is titled "anjuta - Anjuta" and features a menu bar (File, Edit, View, Goto, Project, Build, Tools, Debug, Subversion, Settings, Help) and a toolbar. The central editor shows the file "plugin.c" with the following code:

```
63
64
65  /* is removed from the shell (by the plugin exporting this value), this
66  /* callback will be called.
67  */
68  typedef void (*AnjutaPluginValueRemoved) (AnjutaPlugin *plugin,
69  const char *name,
70  gpointer user_data);
71
72  #define ANJUTA_TYPE_PLUGIN      (anjuta_plugin_get_type ())
73  #define ANJUTA_PLUGIN(o)      (G_TYPE_CHECK_INSTANCE_CAST ((o), ANJUTA_TYPE_PLUGIN, AnjutaPlugin))
74  #define ANJUTA_IS_PLUGIN(o)   (G_TYPE_CHECK_INSTANCE_TYPE ((o), ANJUTA_TYPE_PLUGIN))
75  #define ANJUTA_IS_PLUGIN_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k), ANJUTA_TYPE_PLUGIN))
76  #define ANJUTA_PLUGIN_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o), ANJUTA_TYPE_PLUGIN, AnjutaPluginClass))
77
78  -struct _AnjutaPlugin {
79  GObject parent;
80
81  /* The shell in which the plugin has been added */
82  AnjutaShell *shell;
83
84  /*< private >*/
85  AnjutaPluginPrivate *priv;
86  };
87
88  -struct _AnjutaPluginClass {
89  GObjectClass parent_class;
```

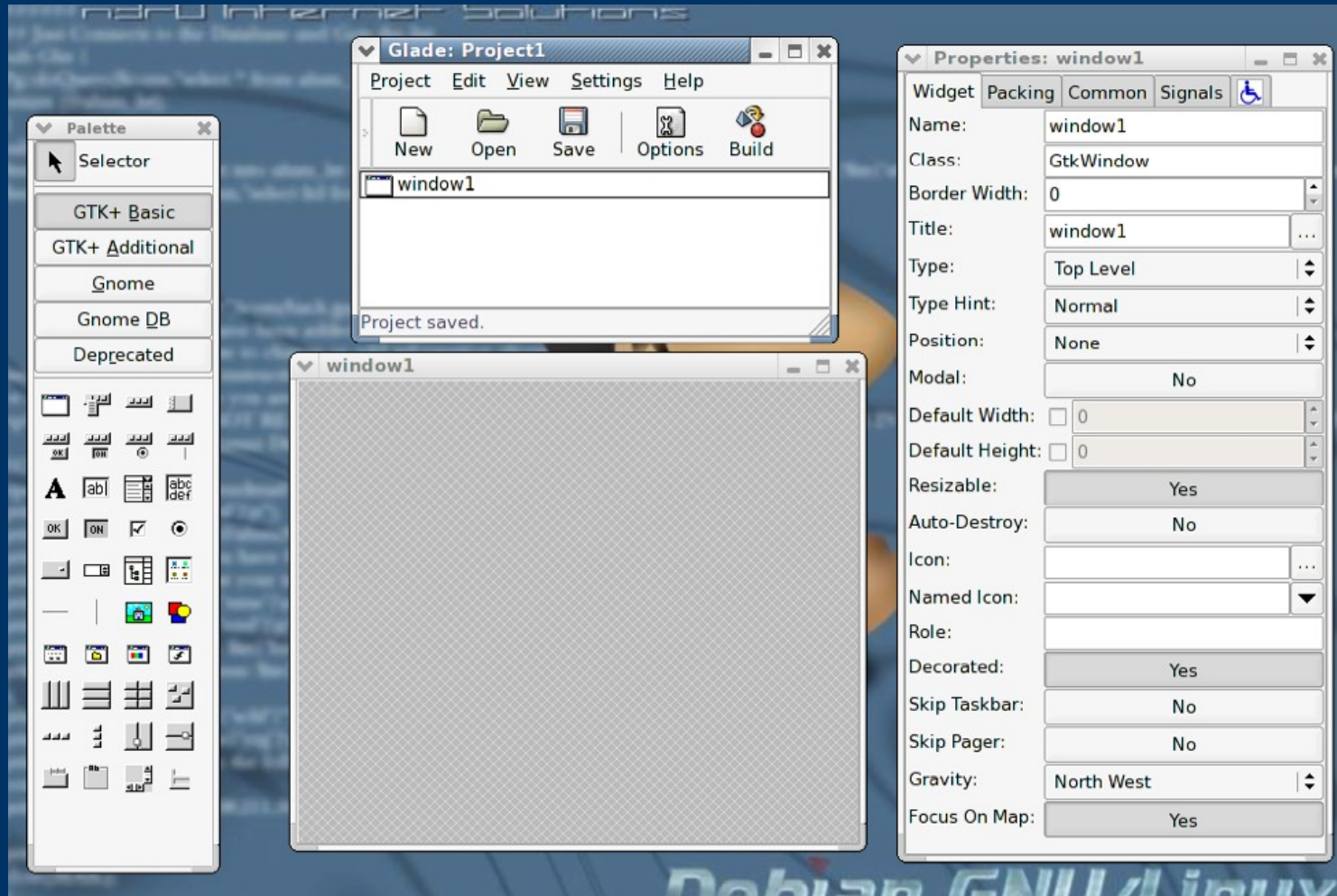
The left sidebar contains a "Files" pane showing the project structure, a "Symbols" pane listing classes like "Accessor", "Action", and "AnEditor", and a "Messages" pane showing build output:

```
Build 1: terminal
Building in directory: /home/naba/projects/anjuta/plugins/editor
make text_editor_prefs.o
Compiling text_editor_prefs.c -> text_editor_prefs.o
Completed... successful
Total time taken: 1 secs
```

The right sidebar shows a "Project" pane with a tree view of files and directories, including "libanjuta-interfaces.la", "libanjuta.la", "test-multi-drag", "test-tree-utils", "test-actions", "test-union", "anjuta-tags", "tm_global_tags.c", "anjuta_launcher", "anjuta", "anjuta-shell", "test_tm_buffer", "libanjuta-ctags.la", "libanjuta-cvs-plugin.la", "gdbmi-test", "libanjuta-gdb.la", "debugger.c", "debugger.h", "gdbmi.c", "gdbmi.h", "plugin.c", "plugin.h", "utilities.c", "utilities.h", "libanjuta-project-import.la", "libanjuta-macro.la", "libanjuta-patch.la", and "test_class_symbol_parsing.la".

The status bar at the bottom indicates: Col: 000 Line: 0076 Project: anjuta Mode: INS Zoom: 0

GLADE



Kdevelop

- Orientado a QT/KDE
 - Gran soporte de C++
 - Soporta C, Java, Pascal, PHP, Ruby, etc
 - Integración con autotools, qmake y ant
 - Ligerero
-
-

Kdevelop (II)

myproj - file:///home/andreas/KDE-work/qm_projects/myproj/src/myproject.cpp - KDevelop

File Edit View Project Build Debug Scripts Bookmarks Window Tools Settings Help

MyProject:MyProject()

```
25 #include <QTextEdit>
26 #include <QTextStream>
27 #include <QCloseEvent>
28 #include <QFileDialog>
29
30 #include "staticlib.h"
31 #include "sharedlib.h"
32
33 MyProject::MyProject()
34 {
35     textEdit = new QTextEdit;
36
37     CSharedLib shLib;
38     double fVal = shLib.GetVal(3.0);
39     QString str;
40     str.setNum(fVal);
41     textEdit->setHtml(str);
42
43     setCentralWidget(textEdit);
44
45     textEdit->
```

Container: QTextEdit
Kind: Function
Access: unknown
File: /tmp/kde-andreas0vjRA/kdevelopV5bga.tmp
Line: 40273 Column: 13

<constructor> QTextEdit(QWidget* parent)
<constructor> QTextEdit(const QString& text, QWidget* parent)
void setReadOnly(bool ro)
void setTextInteractionFlags(Qt::TextInteractionFlags flags)
void setTabChangesFocus(bool b)
void moveCursor(QTextCursor::MoveOperation operation, QTextCursor::MoveMode mode)
void setDocumentTitle(const QString& title)
void setUndoRedoEnabled(bool enable)
void setLineWrapMode(LineWrapMode mode)
<destructor> ~ QTextEdit()

andreas@morpheus:~>cd /home/andreas/KDE-work/qm_projects/myproj
andreas@morpheus:~/KDE-work/qm_projects/myproj>

Application Diff Messages Replace Konsole Find in Files Valgrind Security Problems CTags Breakpoints Problems

Line: 45 Col: 27 INS NORM *

Vim

- Editor con muchas posibilidades
 - Soporta virtualmente cualquier lenguaje
 - Únicamente para editar código
 - Obliga a conocer autotools, etc
 - Ideal para máquinas muy modestas
-
-

Vim (II)

```
file.c [G:operations.c][7:dialogs.h][8:file.h][9:dialogs.c][10:CHANGELOG]
MiniBufExplorer
// reset_done (reportpipe);
// nroffiles = g_list_length (files);
while (cur && op_stop != 1) {
    char *fullpath = malloc (sizeof(char) * (strlen(cur->data)+strlen(srcpath)
    + 2));
    int error = 0;
    report_progress (reportpipe, fullpath, NULL, (int) ((float)1/nroffiles+100));
    if (strcmp(cur->data, ".") != 0) {
        /* XXXX : fullpath function? */
        sprintf (fullpath, "%s/%s", srcpath, (char *)cur->data);
        printf ("%fullpath %s", fullpath);
        i++;
        switch (file_type_withlink(fullpath)) {
            case FTTYPE_NORMAL : error = chown(fullpath, uid, gid); break;
            case FTTYPE_DIR :
                if ((options & OPT_CHOWN_RECURSE) == OPT_CHOWN_RECURSE) {
                    printf ("%filelist_chown: recurse\n");
                    error = dir_chown(actionpipe, reportpipe, fullpath, uid, gid);
                }
                error = file_chown (fullpath, uid, gid);
                break;
            case FTTYPE_LINKNORMAL : error = file_chown(fullpath, uid, gid); break;
            /* XXXX : default: unknown file? */
        }
        report_progress (reportpipe, fullpath, NULL, (int) ((float)1/nroffiles+100));
        switch (error) {
            case -1:
                report_error (reportpipe, fullpath, NULL);
                if (action_read(actionpipe) == 1) {
                    op_stop = 1;
                    break;
                }
            case 1:
                op_stop = 1;
                break;
            default:
                break;
        }
        free (fullpath);
    }
    cur = cur->next;
}
report_done (reportpipe);

dialogs.c (/storage/proj
Tag List 39,5 222
* Press ? for keyboard shortcut
* Sorted by name (lsort) --swp
* = /storage/projects/nimf/cont
../
nifs0086801700000006
CHANGEL0G
COPVING
INSTALL
Makefile
README
RELEASENOTES
TODO
bookmarks.c
bookmarks.h
cmdline.c
cmdline.h
config.c
config.h
context.c
dialogs.c
dialogs.c-backup
dialogs.h
file.c
gtkaddons.c
gtkaddons.h
logo.h
nimf
nimf.c
nimf.h
operations.c
operations.h
panel.c
panel.h
settings.c
settings.h
susinfo.c
knts/nimf [R0] 4,1 Top [Error List]

file.c 1282,1-4 Buf CHANGELOG 6,1 Top
cc nimf.o susinfo.o panel.o cmdline.o file.o operations.o dialogs.o config.o bookmarks.o settings.o gtkaddons.o -o nimf -s -Wall 'gtk-config --cflags --libs' 'lib-config --libs gthread'
```

Emacs

- Editor altamente extensible
- Soporta cualquier lenguaje
- Integración con sistemas de control de versiones, compiladores, debuggers, etc
- Es un IDE sin ser IDE



Emacs

```
ADPAgentC.java

public class ADPAgentC extends ADPAgentC {
  Random r;

  int[] fixedpolicy;
  double[] utilities;
  double[] reward;

  int numStates;
  int numActions;
  double discount;

  double[][][] TransModel;
  double[][] Nsa;
  double[][][] Nsasp;

  int[] statevisited;
  int prevState;
  int prevAction;

  PolicyEvaluation policyEval;

  public ADPAgentC(Mdp mdp) {
    super(mdp);

    this.r = new Random();

    this.numStates = mdp.numStates;
    this.numActions = mdp.numActions;
  }
}

public class ADPAgentC extends ADPAgentC {
  Random r;

  int[] policy;
  double[] utilities;
  double[] reward;

  int numStates;
  int numActions;
  double discount;

  double[][][] TransModel;
  int[][] Nsa;
  int[][][] Nsasp;

  int[] statevisited;
  int prevState;
  int prevAction;

  PolicyEvaluation policyEval;

  ExplorationFunction explorationFunction;

  final int threshold = 5;
  final double unexploredValue = 10000.0;
}

A: --:%% !home!phil!school!6.825!proj3! B: --:-- ADPAgentC.java 4% of 3.4k (
```

Otros recursos

- SCM: control de versiones
- Buscando software
- Alojamiento
- Licencias



SCM

- “Gestión del código fuente”
 - “Control de versiones”
 - SVN, el “bueno”
 - git, arch, bazaar, los “feos”
 - CVS, el “malo”
-
-

Subversion (SVN)

- Desarrollo centralizado
- Sencillo
- “Compatible” con comandos del CVS
- Soporta directorios
- El nuevo estándar para los habituados al CVS



CVS

- Desarrollo centralizado
- Sencillo
- Muy estable
- Un “estándar”
- Obsoleto



SCM's distribuidos

- git
- arch
- bazaar



git

- sencillo
- complejo
- potente
- flexible
- escalable



Buscando software

- Freshmeat
 - Sourceforge
 - Directorio del proyecto GNU
 - Savannah
 - Proyecto Debian
 - GNOME, KDE, ...
-
-

Alojamiento

- Savannah
- Sourceforge



Licencias

- GPL “todos somos iguales ante el código”
- LGPL “somos iguales excepto los desiguales”
- BSD / MIT X11 “somos iguales los que queremos serlo”



Colaborando

- Elegir un proyecto
 - Elegir un módulo o componente
 - Elegir un tarea o bug
 - Desarrollar la solución
 - Exponerla a los miembros del proyecto
 - Reimplementar
-
-

Elegir un proyecto

- Según lenguaje de programación
 - Según la licencia
 - Según el tamaño
 - Según la ayuda y la comunidad creada
-
-

Elegir un módulo

- Según el nivel de cada uno
- Sencillez para empezar
- Pensar funcionalidades no presentes



Elegir una tarea

- Bugzilla
- TODO
- Roadmap
- Bounties (€)



Implementar solución

- Seguir las convenciones y guías de estilo
- Realizar los cambios mínimos necesarios en el código de los demás
- El software libre no es solo Linux
- Considerar la licencia del proyecto



Mostrar el trabajo

- Mostrar el trabajo a la comunidad
 - Buscar la aceptación del mantenedor del proyecto
 - Modificar y reimplementar todo en función de la respuesta
 - Trabajo colaborativo
-
-

Reimplementar

- Aceptar las críticas
- Argumentar las decisiones
- Buscar la excelencia técnica
- Y finalmente, conseguir que acepten el parche



Fin



Licencia

Creative Commons
Reconocimiento-No comercial-Compartir bajo la
misma licencia 2.5 España

