Gestión de Proyectos de SL

INGENIERÍA DE SOFTWARE DE FUENTES ABIERTAS/LIBRES

AMURA FEDERICO-BACINELLO FRANCO-FONZALIDA SANTIAGO-CARBALLO EMMANUEL-DAVID VIRGOLINI-ROSALES MARCIO
GRUPO N.3
CURSO 5K4_

¿Qué es una “comunidad” de SL? (Cont.)

Todas tienen en común:

- El espíritu cooperativo
- La búsqueda continua del mejoramiento y difusión del software libre y del conocimiento
- Su principal interés es la libertad de los usuarios.
- Hay un dominio o interés compartido que le da identidad
- La comunidad es creada y mantenida a partir de las interacciones determinadas, por ejemplo, por actividades conjuntas o discusiones
- Existe una práctica compartida, por ejemplo a través del intercambio de buenas prácticas o lecciones aprendidas.

¿Quiénes pertenecen a una comunidad?

Las personas que forman parte de una comunidad de SL pueden son usuarios, desarrolladores, distribuidores, los que dan soporte, traductores entre otras cosas. Las comunidades pueden abarcar todas estas áreas, o enfocarse en algunas específicas.

La gente que dedica su tiempo al software libre en líneas generales se puede afirmar que es un varón joven con estudios universitarios (o en vías de conseguirlos). La relación del mundo del software libre con la universidad (estudiantes y profesores) es muy estrecha, aunque sigue predominando el desarrollador que no tiene que ver nada con el ámbito académico. Las motivaciones de los desarrolladores están centradas en compartir y aprender.

¿Qué Roles existen?

Usuarios

Contribuyentes

Mantenedor

¿Como se ingresa?

¿Qué te gustaría hacer?

¿Cuánto tiempo dispones?

¿Cómo se se gestiona una comunidad de SL?

Estilos de catedral vs bazar

- Cuidadosamente ensamblados por trabajadores en un espléndido aislamiento, sin que hubiera lugar al lanzamiento de versiones de prueba antes de que hubiera llegado el momento.
- Bazar bullicioso con diferentes agendas y enfoques del cual solo parecía posible que emergiera un sistema coherente y estable mediante una sucesión de milagros.

Estilo de desarrollo de Linus Torvalds

- Lanzar versiones de prueba enseguida y a menudo
- Delegar cuanto sea posible
- Hacer crecer la comunidad
- Estar abierto al cambio

Hacer crecer la comunidad

- Tratar a tus usuarios como colaboradores es el camino menos complicado para mejorar con rapidez y depurar eficazmente un programa.

- Dada una base lo suficientemente amplia de probadores y colaboradores, casi todos los problemas se identificarán con rapidez y su solución será obvia para alguien.

- Un número mayor de usuarios encuentra más errores debido a que añade muchas más formas diferentes de forzar el programa.

Libere pronto, Libere a menudo. Y escucha a los usuarios

- Minimizar la redundancia de esfuerzos mediante la difusión rápida de correcciones ya realizadas.
- Continuo estímulo y una recompensa constante a los usuarios
- Constante revision en cada publicacion. Con el potencial de detectar los errores de forma temprana.
- Menor costo en caso de error.

Delegar cuando sea posible

Aquel que dependa tan sólo de su cerebro al desarrollar un sistema va estar siempre en desventaja frente al que sepa cómo crear un ambiente abierto y en evolución, en el cual la búsqueda de errores y las mejoras se confíen a cientos de personas.

“Soy básicamente una persona muy perezosa a la que le gusta recibir los laureles de lo que otros han hecho”

Linus Torvals

Contexto social de las comunidades de Software Libre

Una comunidad de software libre se asemeja en muchos aspectos a un mercado libre o un sistema ecológico.
Una colección de agentes autónomos que intentan maximizar la utilidad en un proceso que termina en un orden derivado de la autocorrección mucho más eficiente y elaborado de lo que hubiera podido lograr cualquier cantidad de planificación.

Para trabajar y competir con eficacia, programadores que quieran desarrollar un proyecto en colaboración deben aprender a reclutar y motivar a la gente en base a intereses comunes.

Una forma de conectar la individualidad de los colaboradores tan firmemente como fuera posible para llevarla a culminar objetivos difíciles solo alcanzables mediante una colaboración sostenida.

Esto no quiere decir que la visión y la brillantez individual ya no importan. Los proyectos más trascendentales en el mundo del software abierto serán los de aquellos que comiencen a partir de la visión y la brillantez individual, y amplifiquen su importancia mediante la construcción eficaz de grupos con intereses comunes.

La capacidad del coordinador para crear diseños de excepcional brillantez no resulte crítica en una comunidad de software libre, pero sí lo es su capacidad para reconocer las colaboraciones de los demás.

¿Cómo se toman las decisiones?

Nivel de impacto.
Experiencia de los miembros de la comunidad.
Diferentes áreas.

¿Cómo se comparte código e información?

El desafío radica en hacer sentir a todos los miembros de una comunidad distribuida que están trabajando juntos en una misma oficina.
Para tal fin, la mayor parte de los proyectos de software libre de código abierto ofrecen como mínimo un conjunto de herramientas entre las que se encuentran:

¿Cómo se comparte código e información? (Cont.)

- Sitio Web: para transmitir la información del proyecto al público en general;
- Listas de correo electrónico: foro de comunicación dentro del proyecto por excelencia y también el medio registral;
- Control de versiones: para administrar los cambios en el código fuente, incluyendo la posibilidad de revertir los mismos;

¿Cómo se comparte código e información? (Cont.)

- Administrador de bugs: permite a los desarrolladores llevar registro de las actividades en las que están trabajando;
- Chat en tiempo real: un medio para discusiones rápidas e intercambios del tipo pregunta/respuesta.

¿Qué es un “fork”?

Una bifurcación (fork en inglés), en el ámbito del desarrollo de software, es la creación de un proyecto en una dirección distinta de la principal u oficial tomando el código fuente del proyecto ya existente.

Herramientas de gestión

Sitio web

E-mail, chat, foros, blogs.

Repositorio de código, control de versiones.

Gestión de desarrollo/fallos.