

Desarrollo Comunitario

Martín Marqués

2ndQuadrant

25 de abril de 2015

- 1 Formas de participación
- 2 Organización
- 3 Núcleo de desarrollo
- 4 Testers & Reviewers
- 5 Ayuda
- 6 ¿Y de que vivimos mientras tanto?
- 7 Conclusiones
- 8 ¿Preguntas?

- Proponer ideas innovadoras o características que faltan

Formas de participación

- Proponer ideas innovadoras o características que faltan
- Compartir código para implementar dichas características

Formas de participación

- Proponer ideas innovadoras o características que faltan
- Compartir código para implementar dichas características
- Escribir código para pruebas

Formas de participación

- Proponer ideas innovadoras o características que faltan
- Compartir código para implementar dichas características
- Escribir código para pruebas
- Reportar inconvenientes en el uso

Formas de participación

- Proponer ideas innovadoras o características que faltan
- Compartir código para implementar dichas características
- Escribir código para pruebas
- Reportar inconvenientes en el uso
- Compartir parches que solucionan dichos problemas

Formas de participación

- Proponer ideas innovadoras o características que faltan
- Compartir código para implementar dichas características
- Escribir código para pruebas
- Reportar inconvenientes en el uso
- Compartir parches que solucionan dichos problemas
- Participar ayudando en foros y listas

Contenido

- 1 Formas de participación
- 2 Organización**
- 3 Núcleo de desarrollo
- 4 Testers & Reviewers
- 5 Ayuda
- 6 ¿Y de que vivimos mientras tanto?
- 7 Conclusiones
- 8 ¿Preguntas?

Todo tiene un orden, o no funciona

Todo tiene un orden, o no funciona

- Existen estructuras jerárquicas

Todo tiene un orden, o no funciona

- Existen estructuras jerárquicas (principalmente por meritocracia)

Todo tiene un orden, o no funciona

- Existen estructuras jerárquicas (principalmente por meritocracia)
- Ciclos regulares de entrega de nuevas funcionalidades

Todo tiene un orden, o no funciona

- Existen estructuras jerárquicas (principalmente por meritocracia)
- Ciclos regulares de entrega de nuevas funcionalidades
- Compromiso personal:

Todo tiene un orden, o no funciona

- Existen estructuras jerárquicas (principalmente por meritocracia)
- Ciclos regulares de entrega de nuevas funcionalidades
- Compromiso personal: cada uno se hace cargo de su código

Contenido

- 1 Formas de participación
- 2 Organización
- 3 Núcleo de desarrollo**
- 4 Testers & Reviewers
- 5 Ayuda
- 6 ¿Y de que vivimos mientras tanto?
- 7 Conclusiones
- 8 ¿Preguntas?

- Gente involucrada con el proyecto

- Gente involucrada con el proyecto
- Por lo general, ya han hecho valiosas colaboraciones

- Gente involucrada con el proyecto
- Por lo general, ya han hecho valiosas colaboraciones
- Se los considera serios:

- Gente involucrada con el proyecto
- Por lo general, ya han hecho valiosas colaboraciones
- Se los considera serios: no andan comprometiendo código sin analizar, probar y tener consenso en el grupo de desarrolladores

- Gente involucrada con el proyecto
- Por lo general, ya han hecho valiosas colaboraciones
- Se los considera serios: no andan comprometiendo código sin analizar, probar y tener consenso en el grupo de desarrolladores (y la comunidad también)

- Gente involucrada con el proyecto
- Por lo general, ya han hecho valiosas colaboraciones
- Se los considera serios: no andan comprometiendo código sin analizar, probar y tener consenso en el grupo de desarrolladores (y la comunidad también)
- También participan en las listas generales ayudando usuarios finales

Contenido

- 1 Formas de participación
- 2 Organización
- 3 Núcleo de desarrollo
- 4 Testers & Reviewers**
- 5 Ayuda
- 6 ¿Y de que vivimos mientras tanto?
- 7 Conclusiones
- 8 ¿Preguntas?

Probar código y reportar inconvenientes

- Puede ser cualquiera

- Puede ser cualquiera → **podrías ser vos!**

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando (a veces el código no coincide con totalmente con la documentación)

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando (a veces el código no coincide con totalmente con la documentación)
- Al estar disponible el código 2: puede proponer un parche

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando (a veces el código no coincide con totalmente con la documentación)
- Al estar disponible el código 2: puede proponer un parche
- Revisar nuevo código antes que esté disponible en una versión estable

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando (a veces el código no coincide con totalmente con la documentación)
- Al estar disponible el código 2: puede proponer un parche
- Revisar nuevo código antes que esté disponible en una versión estable
 - No requiere de conocimientos de programación

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando (a veces el código no coincide con totalmente con la documentación)
- Al estar disponible el código 2: puede proponer un parche
- Revisar nuevo código antes que esté disponible en una versión estable
 - No requiere de conocimientos de programación
 - Ayuda mucho a la persona que escribió el código

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando (a veces el código no coincide con totalmente con la documentación)
- Al estar disponible el código 2: puede proponer un parche
- Revisar nuevo código antes que esté disponible en una versión estable
 - No requiere de conocimientos de programación
 - Ayuda mucho a la persona que escribió el código
 - Muy difícil de llevar a cabo si el revisor no tiene acceso al código

Probar código y reportar inconvenientes

- Puede ser cualquiera → **podrías ser vos!**
- No requiere de elevados conocimientos de programación → aunque tenerlos ayuda mucho
- Al estar disponible el código, puede ver que está pasando (a veces el código no coincide con totalmente con la documentación)
- Al estar disponible el código 2: puede proponer un parche
- Revisar nuevo código antes que esté disponible en una versión estable
 - No requiere de conocimientos de programación
 - Ayuda mucho a la persona que escribió el código
 - Muy difícil de llevar a cabo si el revisor no tiene acceso al código
- Corregir documentación, o escribir documentación que falta

Contenido

- 1 Formas de participación
- 2 Organización
- 3 Núcleo de desarrollo
- 4 Testers & Reviewers
- 5 Ayuda**
- 6 ¿Y de que vivimos mientras tanto?
- 7 Conclusiones
- 8 ¿Preguntas?

Listas y foros de ayuda

- Ayudar en el uso a usuarios finales

- Ayudar en el uso a usuarios finales
- Proponer ideas en listas de desarrolladores basadas en problemas con los que se encontró

- Ayudar en el uso a usuarios finales
- Proponer ideas en listas de desarrolladores basadas en problemas con los que se encontró
- Organizar juntadas y/o eventos y charlas

- Ayudar en el uso a usuarios finales
- Proponer ideas en listas de desarrolladores basadas en problemas con los que se encontró
- Organizar juntadas y/o eventos y charlas → **¡Como esta!**
- Compartir el conocimiento

- Ayudar en el uso a usuarios finales
- Proponer ideas en listas de desarrolladores basadas en problemas con los que se encontró
- Organizar juntadas y/o eventos y charlas → **¡Como esta!**
- Compartir el conocimiento → lo que hoy das, mañana vuelve

Contenido

- 1 Formas de participación
- 2 Organización
- 3 Núcleo de desarrollo
- 4 Testers & Reviewers
- 5 Ayuda
- 6 ¿Y de que vivimos mientras tanto?**
- 7 Conclusiones
- 8 ¿Preguntas?

Haciendo negocio con Software Libre

- Los proyectos de cero son costosos

- Los proyectos de cero son costosos → pero hay gente dispuesta a pagarlo

- Los proyectos de cero son costosos → pero hay gente dispuesta a pagarlo
- Desarrollos alrededor de proyectos ya existente

- Los proyectos de cero son costosos → pero hay gente dispuesta a pagarlo
- Desarrollos alrededor de proyectos ya existente
- Si por contrato se libera el código pautado, ganamos todos

- Los proyectos de cero son costosos → pero hay gente dispuesta a pagarlo
- Desarrollos alrededor de proyectos ya existente
- Si por contrato se libera el código pautado, ganamos todos → hay que convencer a la otra parte

- Los proyectos de cero son costosos → pero hay gente dispuesta a pagarlo
- Desarrollos alrededor de proyectos ya existente
- Si por contrato se libera el código pautado, ganamos todos → hay que convencer a la otra parte
- Soporte!

- Los proyectos de cero son costosos → pero hay gente dispuesta a pagarlo
- Desarrollos alrededor de proyectos ya existente
- Si por contrato se libera el código pautado, ganamos todos → hay que convencer a la otra parte
- Soporte!
- Cursos de capacitación

- Los proyectos de cero son costosos → pero hay gente dispuesta a pagarlo
- Desarrollos alrededor de proyectos ya existente
- Si por contrato se libera el código pautado, ganamos todos → hay que convencer a la otra parte
- Soporte!
- Cursos de capacitación → hay que leer mucho código primero!

Contenido

- 1 Formas de participación
- 2 Organización
- 3 Núcleo de desarrollo
- 4 Testers & Reviewers
- 5 Ayuda
- 6 ¿Y de que vivimos mientras tanto?
- 7 Conclusiones**
- 8 ¿Preguntas?

Conclusiones

- Cuando escriben código, traten de que vuelva a la comunidad

- Cuando escriben código, traten de que vuelva a la comunidad
- Prueben código que aún no hay sido liberada como estable

- Cuando escriben código, traten de que vuelva a la comunidad
- Prueben código que aún no hay sido liberada como estable
- Participen de eventos de la comunidad de SoftWare Libre

- Cuando escriben código, traten de que vuelva a la comunidad
- Prueben código que aún no hay sido liberada como estable
- Participen de eventos de la comunidad de SoftWare Libre
- Participen en listas de correo y/o foros, tanto de ayuda como de desarrollo

- Cuando escriben código, traten de que vuelva a la comunidad
- Prueben código que aún no hay sido liberada como estable
- Participen de eventos de la comunidad de SoftWare Libre
- Participen en listas de correo y/o foros, tanto de ayuda como de desarrollo
- Den una charla como esta

Contenido

- 1 Formas de participación
- 2 Organización
- 3 Núcleo de desarrollo
- 4 Testers & Reviewers
- 5 Ayuda
- 6 ¿Y de que vivimos mientras tanto?
- 7 Conclusiones
- 8 ¿Preguntas?**

¿Preguntas?