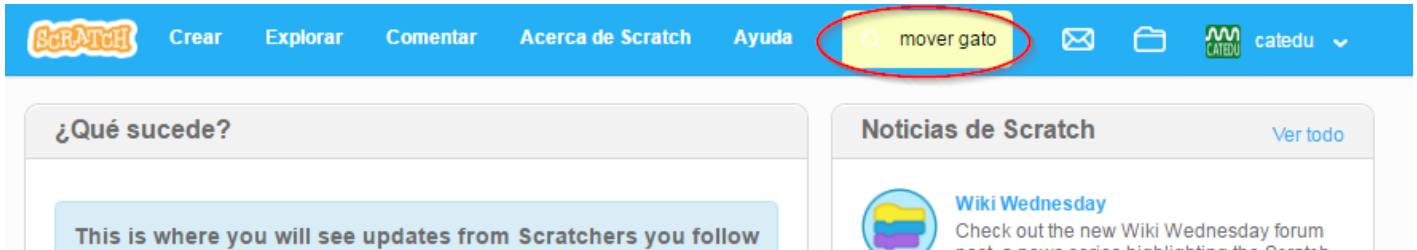


Reinventar

Buscamos un programa que esté relacionado con el movimiento del sprit, por ejemplo:



y encontramos este <https://scratch.mit.edu/projects/38947928/>:

<https://scratch.mit.edu/projects/embed/38947928/?autostart=false>

Nos gusta pero vemos que se podría mejorar:

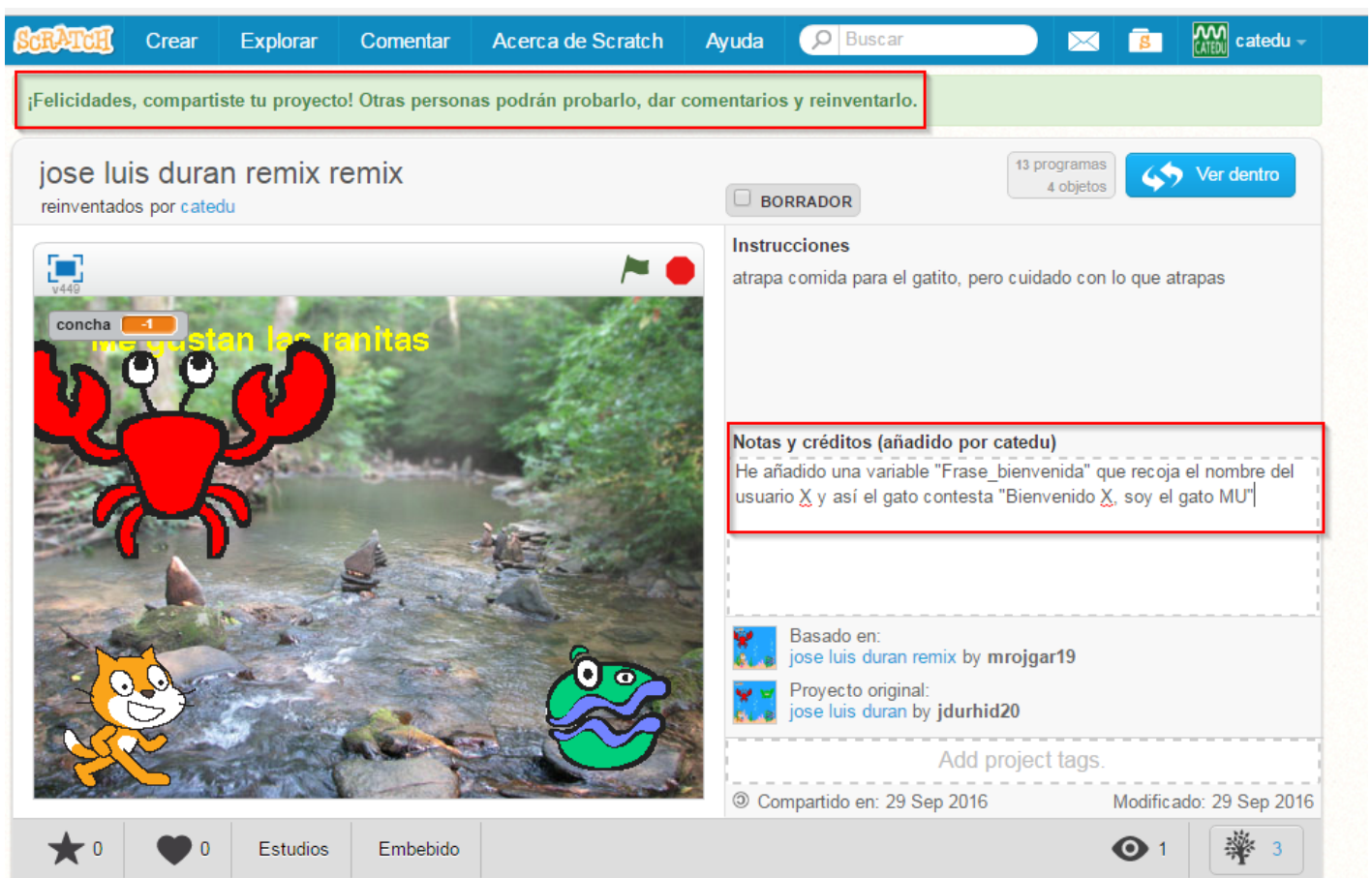


Mejoramos el código, añadiendo una variable "frase_bienvenida" que recoja la respuesta y el texto que queremos que aparezca

Compartir

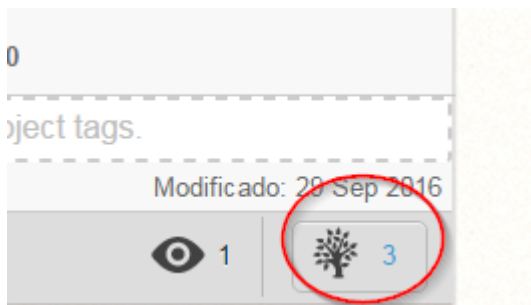


Le damos a compartir y nos sale un mensaje de bienvenida y la opción de describir nuestra mejora:



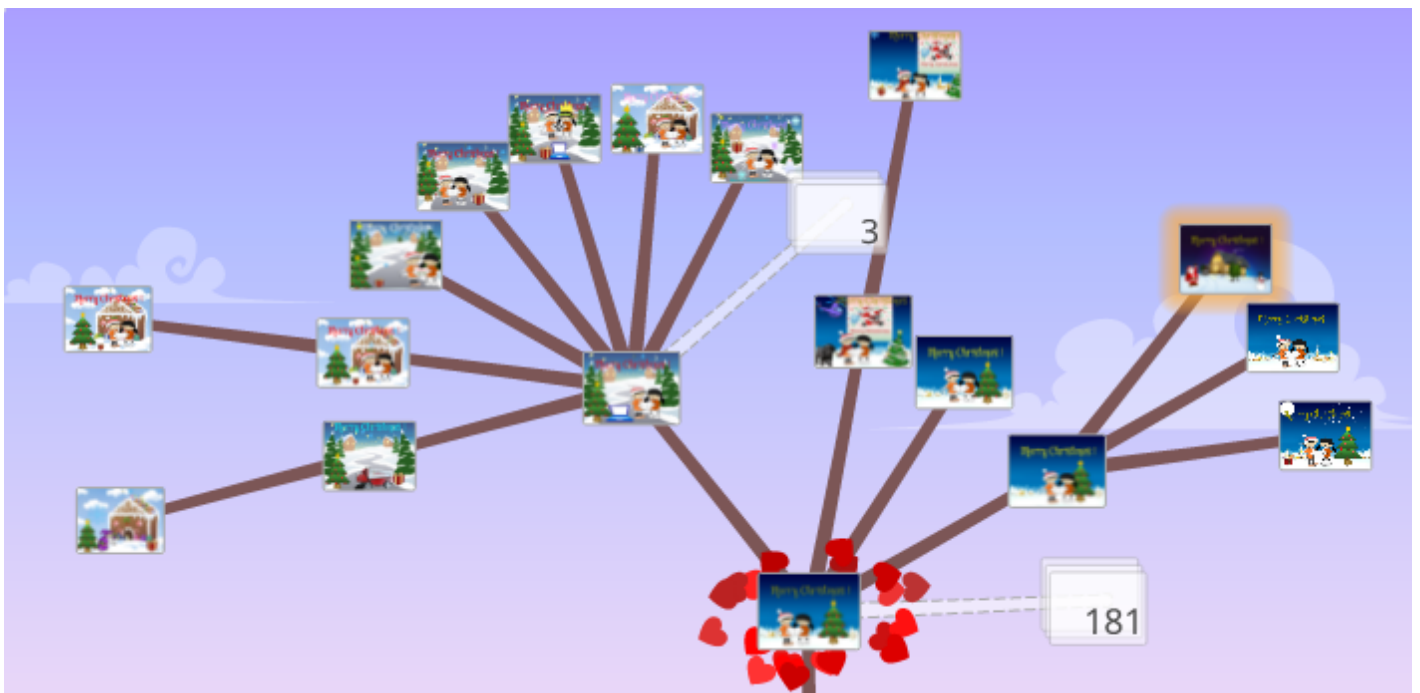
y ahora es otro proyecto: <https://scratch.mit.edu/projects/123302249/>

<https://scratch.mit.edu/projects/embed/123302249/?autostart=false>



Si pinchamos en el árbol sale [esta página](#) y aparece 3 reinversiones (a fecha 29/9/16) el primero es el original, uno segundo (del mismo autor que lo mejoró) y esta última de Catedu

Hay proyectos donde las reinversiones son muchas y salen muchas versiones de la misma, por ejemplo [en este proyecto](#) de tarjetas navideñas [el árbol de reinversiones](#)



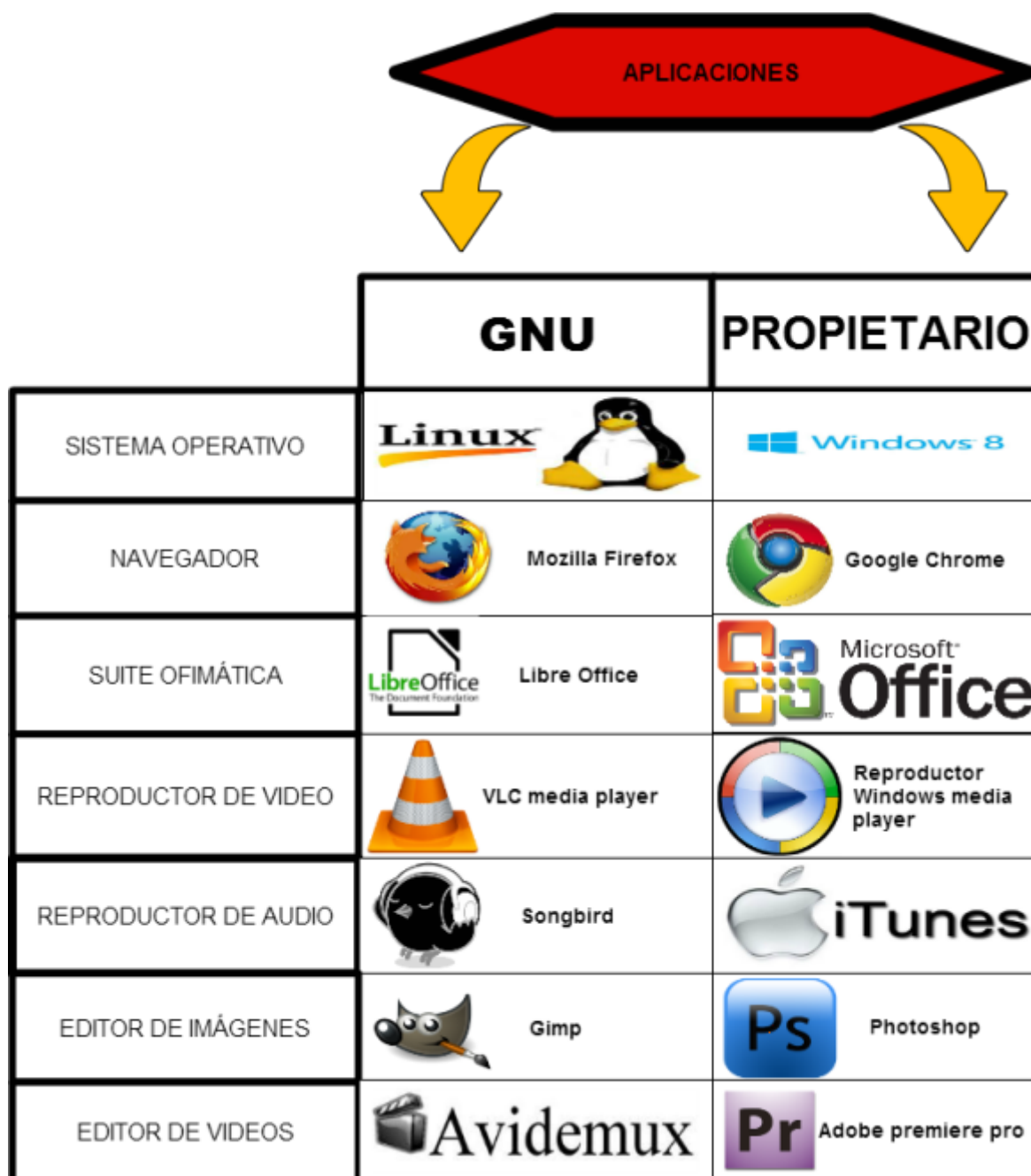
¿Por qué reinventar?

Recordemos que [Scratch es código libre](#), por lo tanto es [software libre](#) de ser copiado, estudiado o modificado. Esto no debe de confundirse con [OpenSource](#) o código abierto, con licencia abierta para que técnicamente se mejore, o [FreeWare](#), es decir gratis.

Aunque técnicamente no se esté obligado a mejorar el código como en OpenSource, sí que se obliga a acceder al código fuente para poder copiarlo y por lo tanto tenemos libertad de reinventar, podemos obtener una versión diferente que no necesariamente nos obligue a una mejora, somos libres de hacer con otros programas de otros usuarios lo que queramos mientras respetemos la licencia libre (acceso al código principalmente)

Se puede aprovechar la actividad de reinventar en el aula para introducir los conceptos de mejora de código comunitarios:

- Los 4 principios filosóficos del software libre. ([ver](#))
- Los 10 principios técnicos del software abierto ([ver](#))
- El software gratis no es libre o abierto y viceversa
- La mejora de la comunidad gracias al acceso de la licencias [GNU o GPL es decir Licencia Publica](#)
- La historia de [GNU/Linux](#)
- Desarrollo de programas libres frente a los privativos



Conviértete en un Scratcher

Al cabo de un tiempo Scratch te preguntará si quieres ser un **Scratcher** ¿qué implica? pues básicamente que respetarás las normas básicas de convivencia un foro de este tipo, actualmente se fija en los siguientes aspectos:

- Limitaciones en tus **comentarios**: Tus críticas tienen que ser constructivas, no negativas y respetando las opiniones, trabajos y comentarios de los demás
- Limitaciones en **tiempo de respuesta** en los debates (dirigidos a evitar spam)
- Limitaciones en el uso de **enlaces externos**, imágenes, almacenamientos en la nube, etc... evidentemente sí que puedes usar enlaces a otros proyectos Scratch

El Scratcher, por supuesto tiene que estar registrado y aunque no es necesario, lo normal es que comparta proyectos, haga comentarios de otros trabajos, reinvente... es decir participar en la comunidad Scratch.

Aquí se puede trabajar los **valores de las comunidades virtuales**, la importancia de los comentarios, foros, debates ... sin perder la vista el **constructivismo, colaboración** ... no la simple palabrería. La cooperación, no sólo técnicamente, sino también se puede ser animar, felicitar ...

https://wiki.scratch.mit.edu/wiki/New_Scratcher_Status

Revision #3

Created 1 February 2022 11:38:02 by Equipo CATEDU

Updated 2 November 2022 20:27:32 by Equipo CATEDU