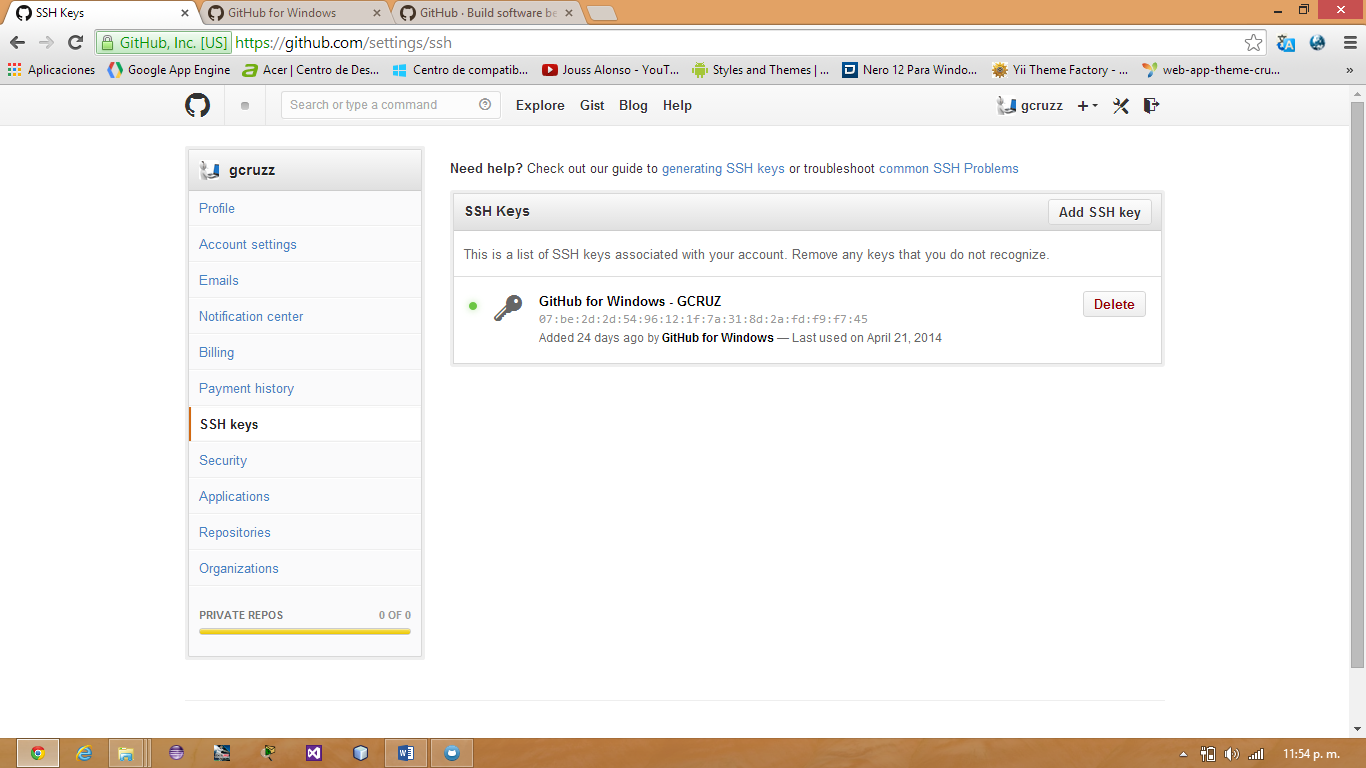
**PASOS PARA CONTRIBUIR AL REPOSITORIO CENTRAL (**[**https://github.com/gcruzz/playclass**](https://github.com/gcruzz/playclass)**)**

**PARA TRABAJAR UN FLUJO DE TRABAJO EQUIVALENTE AL DE SUBVERSION FAVOR SEGUIR ESTOS PASOS**

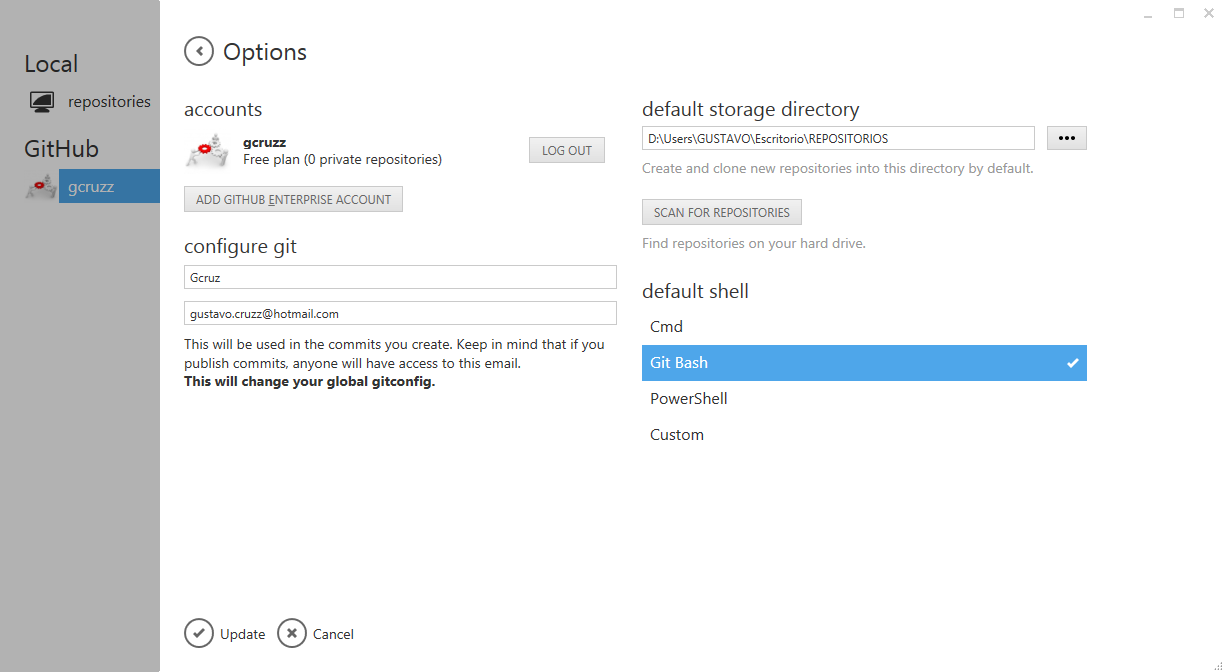
1. Instalar github for Windows.

<https://windows.github.com/>

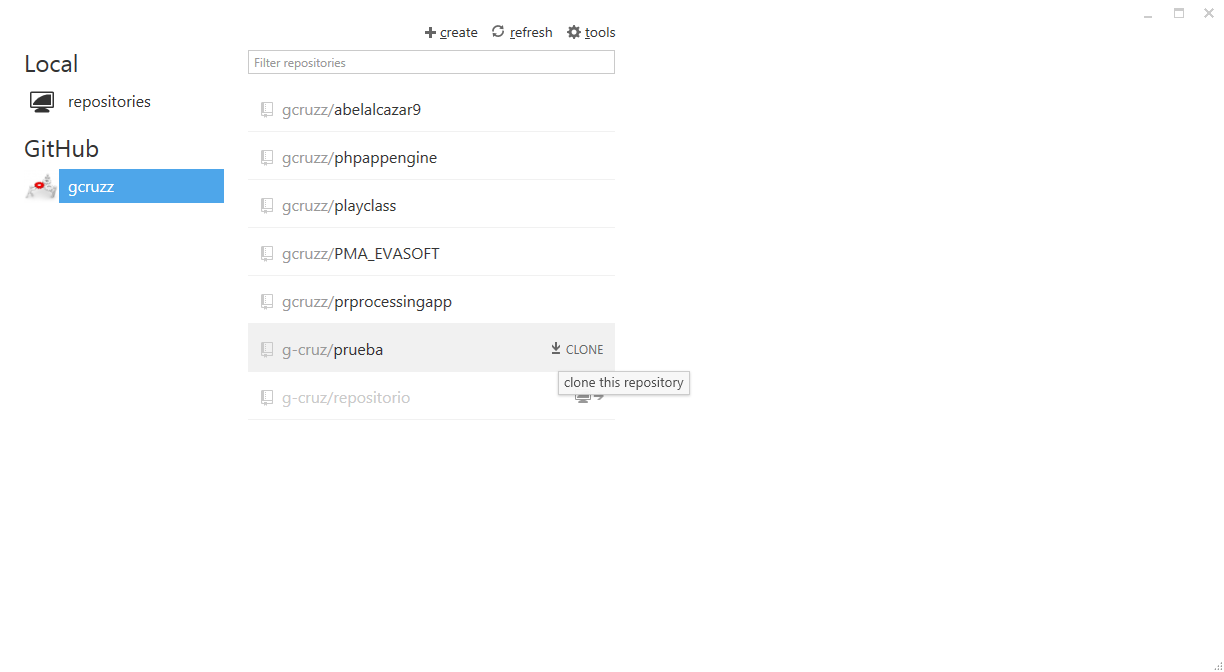
1. Luego ejecutar la aplicación y para sincronizarse con github loguerse con el correo y la clave de github, si no tiene cuenta de github ir a https://github.com/ registrarse y repetir el paso.
2. Una vez logueado la aplicación se sincroniza automáticamente con la cuenta que está en la nube creándole de forma automática una clave ssh para permitir conexión de forma segura, dejando un registro como este:



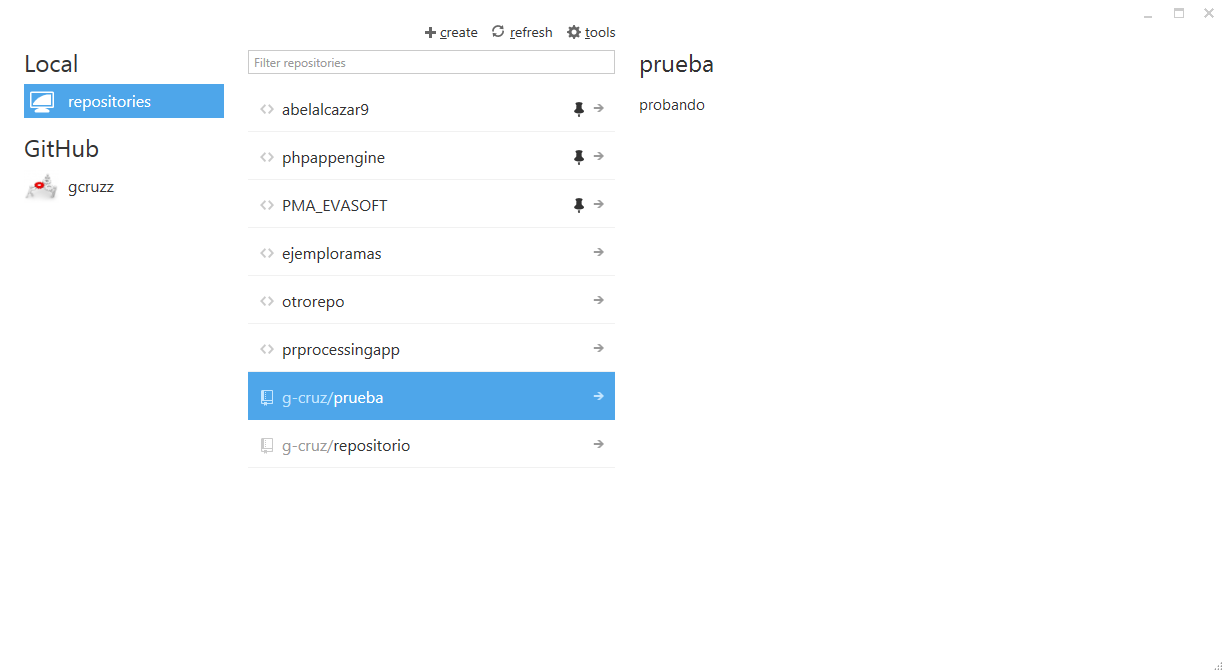
1. Luego definir el directorio donde guardas deseas guardar tus repositorios locales.



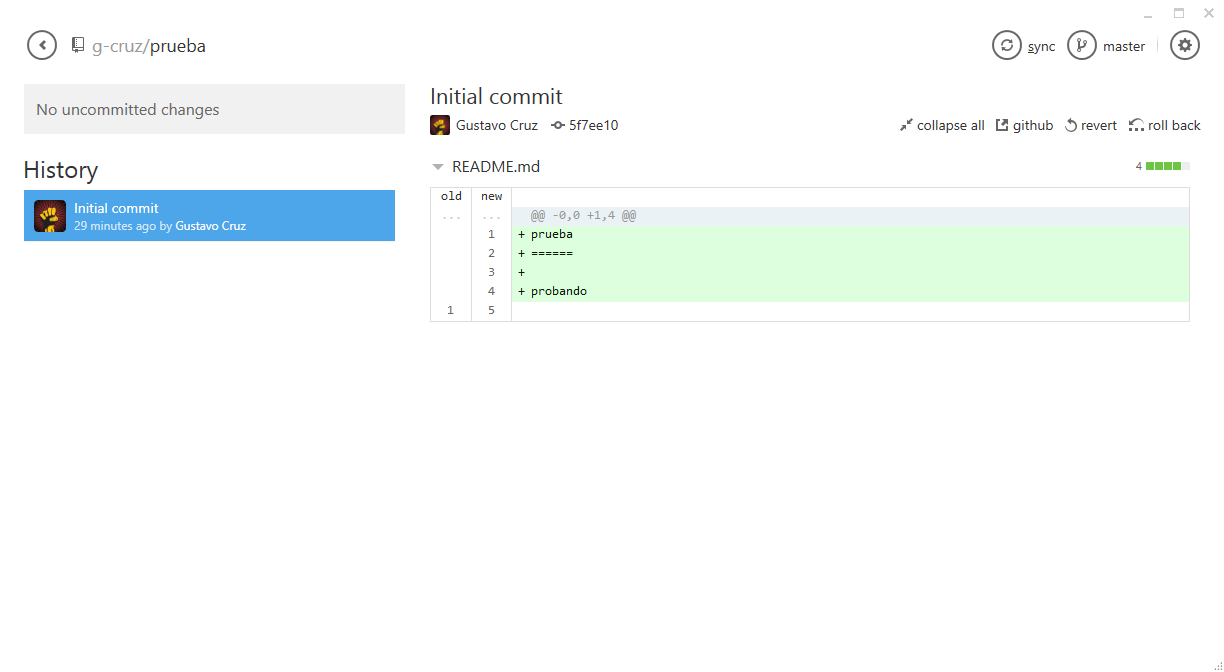
1. Ahora vamos a la pantalla inical donde aparecen los repositorios locales y los repositorios remotos que tengas en tu cuenta de github, pero también se muestran los repositorios a los que tienes permisos para contribuir como aparece en pantalla, en este caso yo tendré que agregarles como cuntribuidores a ese repositorio central para que al darle “refresh” les aparezca y le damos click en “CLONE”, que automáticamente me descarga el proyecto del repositorio central y me lo ubica en el directorio por defecto local de mis repositorios.



1. Una vez clonado ya pueden realizar cambios y vemos que aparece en repositorios locales.



1. Al darle click vemos que aparece el histórico de cambios.



1. Ahora podemos adicionar cambios a nuestros archivos en nuestro proyecto y abrimos la consola gitshell que también se instala con esta aplicación y cuando surgen cambios en nuestro repositorio ejecutamos.

* git status //Esto nos mostrar el estado de los archivos si están en seguimiento o no.
* git add archivo.txt //Para preparar archivos que serán confirmados o que iran en el commit.
* git commit -m "Contribuyendo con estos cambios" //Confirmar cambios locales.
* gitk //visualiza el histórico de confirmaciones a través de GUI de git.
* git help //Recomendable para obtener información de los comandos y para que sirven.

Estos serían los comandos que más se usarían al trabajar localmente. También se pueden aplicar commit a través de la aplicación cliente de github ya que en mi caso por costumbre uso la consola para todo. Ahora la parte de viene la parte de subir los cambios al repositorio central, que necesariamente para trabajar el mismo flujo de trabajo de subversión lo haremos por consola, aunque también se puede por la aplicación es necesario para este caso para mantener un histórico lineal como en subversión con estos comandos:

* git push origin master //Aplicando cambios al servidor origen es decir el central
* git pull –rebase origin master //Traer los cambios del servidor central y los aplica localmente dejándome el histórico lineal, todo esto es equivalente al svn update de subversión

Cuando hay cambios en el servidor central que no tengo localmente si se tiene abierta la aplicación aparecerá en **el botón “sync”** un +”n “siendo n el número de cambios que están en el servidor central pero que no tengo localmente. **Por favor no presionar este botón porque me ejecutaria solamente “git pull origin master” desorganizándome el histórico** y eso es lo que no queremos al menos para este flujo de trabajo, es por eso que debemos adicionarle el –rebase.

**Resumen**

Ya que en git puede manejar flujos de trabajo distribuidos, debido a que es un sistema de control de versiones distribuido, para este caso implementaremos un flujo de trabajo con un repositorio central como en subversión, que varios personas realizan sus actividades y las envían al repositorio central.

La idea es trabajar en el repositorio local, realizando cambios y agregando archivos, confirmando, etc; todo tipo de modificaciones, y cuando ya tengamos algo importante subir los cambios al repo central, pero antes actualizar “git pull –rebase origin master” y luego subir “git push origin master” los commits locales.

Básicamente eso es todo lo necesario, cualquier cosa me la comunican.

Aquí les dejo el libro de git en español para el que quiera profundizar.

<http://git-scm.com/book/es>

Comandos.

<http://elbauldelprogramador.com/mini-tutorial-y-chuleta-de-comandos-git/>