

## **INTEGRACIÓN DE LAS HERRAMIENTAS DE "GITHUB EDUCATION" EN EL AULA**

### **INTEGRATION OF 'GITHUB EDUCATION' TOOLS IN THE CLASSROOM**

Coromoto León

[cleon@ull.edu.es](mailto:cleon@ull.edu.es)

Gara Miranda

[gmiranda@ull.edu.es](mailto:gmiranda@ull.edu.es)

Casiano Rodríguez

[criguezl@ull.edu.es](mailto:criguezl@ull.edu.es)

Eduardo Segredo

[esegredo@ull.edu.es](mailto:esegredo@ull.edu.es)

Universidad de La Laguna, España

Carlos Segura González

[c.segura@cimat.mx](mailto:c.segura@cimat.mx)

Centro de investigación en Matemáticas- CIMAT, México

## RESUMEN

El sistema de control distribuido de versiones Git se ha convertido en el estándar de facto para manejar proyectos software. Uno de los motivos de la creciente popularidad de Git es el éxito de GitHub, una plataforma Web de desarrollo colaborativo. GitHub ofrece toda la funcionalidad de Git e integra diversas herramientas de control de acceso, colaboración, gestión de tareas y control de proyectos, todo ello en la nube. Educadores tanto del mundo académico relacionado con la Informática como de fuera de ella utilizan GitHub en sus cursos. En este trabajo se presenta cómo se han utilizado las herramientas Git, GitHub y GitHub Classroom para gestionar la parte práctica de varias asignaturas de los estudios del grado en Ingeniería Informática. Esta contribución se centra en motivar esta experiencia, explicar su implementación y discutir los resultados obtenidos.

**PALABRAS CLAVE:** Aprendizaje colaborativo, control de versiones, codificación social

## ABSTRACT

The distributed version control system Git has become the de facto standard for managing software projects. One of the reasons for the growing popularity of Git is the success of GitHub, a collaborative development Web platform. GitHub offers all the functionality of Git and integrates various tools for access control, collaboration, task management and project control, all of them in the cloud. Educators from both the academic world related to IT and from outside, are using GitHub in their courses. This paper presents how the Git, GitHub and GitHub Classroom tools have been used to manage the laboratories of several subjects of the degree studies in Computer Engineering. This contribution focuses on motivating this experience, explaining its implementation and discussing the results obtained.

**KEYWORDS:** collaborative learning, version control, social coding

## INTRODUCCIÓN

La razón por la que los desarrolladores de software viven a la vanguardia de un futuro distribuido es que sus herramientas de trabajo siempre han sido artefactos digitales, y desde el comienzo de las redes, su proceso de trabajo les obliga a permanecer conectados. Las herramientas con las que los desarrolladores de software trabajan y las culturas que rodean el uso de esas herramientas tienden a popularizarse en la sociedad. En retrospectiva, ahora parece obvio que el correo electrónico y la mensajería instantánea hayan llegado a las masas, cuando ambos fueron utilizados por los desarrolladores de software antes que nadie. Es menos obvio que Git (Chacon y Straub, 2014) un sistema de control de versiones inventado para coordinar el desarrollo del kernel de Linux y GitHub (GitHub Press, 2017) una plataforma Web de desarrollo colaborativo para alojar proyectos utilizando el sistema de control de versiones Git vayan a popularizarse de la misma manera. La mayoría de la gente no crea código para ganarse la vida. Pero a medida que los productos

y procesos de trabajo de cada profesión se digitalizan cada vez más, al final se harán necesarias herramientas que permitan coordinar el trabajo en equipo. Es por eso que Git ya se utiliza para administrar el desarrollo colaborativo de recetas, listas musicales, libros, fuentes, documentos legales, lecciones y tutoriales, conjuntos de datos, etc. ¿Cómo ha podido producirse esta expansión dada la complejidad de Git? La razón principal es el desarrollo de GitHub como red social. En GitHub se «narra tu trabajo» se hace el «trabajo observable» hay una «colaboración abierta». Así pues, educadores de dentro y fuera del mundo académico relacionados o no con la Informática han comenzado a usar las herramientas basadas en Git en sus cursos.

En este trabajo se presenta cómo se han utilizado las herramientas Git, GitHub y GitHub Classroom para gestionar asignaturas. En concreto se ha utilizado en las asignaturas de seis créditos Lenguajes y paradigmas de programación (Rodríguez, León, Miranda, Segredo y Segura, 2013), Desarrollo de Sistemas Informáticos, Sistemas y Tecnologías Web y Procesadores de lenguajes de los estudios de grado en Ingeniería Informática de la Escuela Superior de Ingeniería y Tecnología de la Universidad de La Laguna. A lo largo de las asignaturas el alumno debe desarrollar de manera independiente o en equipos las prácticas de laboratorio. En todas las asignaturas el sistema de evaluación incluye un porcentaje del 40% de evaluación continua. Para gestionar esta parte de la evaluación se han utilizado las herramientas que se describen en este trabajo.

El resto del trabajo se organiza de la siguiente manera: En primer lugar se describen qué son Git y Github. En las siguientes secciones se muestra cómo configurar una clase de GitHub Classroom por parte del profesor y a continuación debe usarla un estudiante. Finalmente, se encuentran los epígrafes de discusión y las conclusiones.

## ¿QUÉ ES GIT?

Git es un sistema de control de versiones distribuido. Los Sistemas de Control de Versiones (SCV) son herramientas de uso habitual en el desarrollo profesional de software desde hace décadas. Proporcionan una serie de funcionalidades claves para el desarrollo de proyectos como son: el control de cambios en el código, la reversibilidad de dichos cambios, y la posibilidad de colaborar en el desarrollo del código. Además, los SCV permiten tener en paralelo varias versiones o ramas del proyecto. Las ramas se utilizan para desarrollar funcionalidades aisladas de los cambios en otras partes del proyecto que posteriormente pueden integrarse en la rama principal. Los SCV se pueden clasificar en dos grandes familias: SCV centralizados y SCV distribuidos. Los SCV centralizados son sistemas cliente-servidor donde hay un repositorio canónico en el servidor que contiene toda la información de los cambios mientras que los clientes solo

tienen copias de trabajo. En un sistema distribuido no existe el concepto de repositorio canónico por lo que cada cliente ha de tener una copia completa del repositorio.

Según López-Pellicer, Béjar y Latre (2015) algunas de las razones por las que se están incorporando a la enseñanza los SCV son: ayudar a desarrollar la competencia de trabajo en equipo, facilitar la retroalimentación de los alumnos, establecer escenarios de desarrollo realistas e, incluso, poder utilizarse como herramienta de evaluación. En este trabajo se ha utilizado Git como un sistema de desarrollo de las prácticas de laboratorio aprovechando sus funcionalidades. Sin embargo, como su curva de aprendizaje es elevada se han incluido prácticas tutorizadas para adquirir destreza en su uso de manera que no afecten al desarrollo normal del curso. Esta decisión, aumenta las tareas de gestión docente por lo que se ha optado por utilizar GitHub Classroom. León, Miranda, Rodríguez y Segredo (2015) utilizan la plataforma y las herramientas de escritorio proporcionadas por GitHub para el desarrollo de la parte práctica de un curso genérico en el que no se contaba con estudiantes del grado en Ingeniería Informática por lo que no era aconsejable utilizar línea de comandos.

## ¿QUÉ ES GITHUB?

GitHub es un servicio comercial que permite el alojamiento y desarrollo de repositorios con control de versiones y mecanismos de Colaboración para el desarrollo de proyectos conjuntos desde cualquier sitio. GitHub proporciona una interfaz Web que permite al usuario registrado crear repositorios vacíos o por clonación de otro repositorio hospedado en GitHub (*fork* en la nomenclatura de GitHub), enviar solicitudes de cambio en repositorios hospedados (*pull request*), y gestionar dichas solicitudes. Además, los repositorios hospedados en GitHub pueden actuar como repositorios remotos de repositorios locales. Los repositorios que se crean en GitHub son por defecto de acceso público. Solo mediante una cuenta de pago o si cumplen ciertas condiciones es posible alojar repositorios privados. Cada repositorio de GitHub es propiedad de una cuenta de usuario o de organización. Las modificaciones a un repositorio alojado en GitHub solo se pueden realizar por usuarios que han iniciado sesión y que estén autorizados a modificar su contenido, o a aquellos autorizados que han configurado un repositorio local de Git para que con las credenciales proporcionadas por GitHub pueda conectarse a dicho repositorio. Además del alojamiento, GitHub proporciona a cada repositorio una wiki, un gestor de tareas (*issues*), un completo sistema de gestión de comentarios, un cuadro de control con grafos sociales e, incluso, una página web propia.

En el campo educativo GitHub ha creado cuentas de usuario especiales para profesores, estudiantes y centros educativos, agrupadas en el proyecto GitHub Classroom (Griffin y Seals, 2013). Con estas herramientas

es posible crear repositorios conjuntos para alumnos y profesores de manera que todos puedan participar en la realización de un mismo proyecto o ejercicio. Así, se mejora la interacción entre alumnos y profesores para la revisión de código ante posibles mejoras y/o errores. Todo esto administrado bajo una única cuenta, que será obviamente la del tutor o profesor.

GitHub Classroom automatiza la creación de repositorios y el control de acceso a los mismos, lo que facilita a los profesores la distribución de los distintos ejercicios y la recogida de retroalimentación por parte de los estudiantes. El núcleo de GitHub classroom son las asignaciones de tareas. Los profesores crean un ejercicio que se distribuirá a los estudiantes utilizando un enlace de invitación. Estos ejercicios o trabajos podrán ser realizados por los estudiantes de manera individual, o bien en grupos, junto con los profesores en un repositorio conjunto.

GitHub cuenta también con el programa orientado a estudiantes ‘Student Developer Pack’, un paquete de inicio para programadores novatos, en el que se incluyen las mejores herramientas de desarrollo, según su juicio, para que aprendan usándolas.

The screenshot shows a GitHub repository page for 'ULL-ESIT-SYTW-1617/presentaciones-todos'. The repository has 204 commits, 2 branches, 0 releases, and 18 contributors. The commit history shows a recent commit by Daniel on 23 Dec 2016, titled 'Arreglando el script y generando el indice'. The file list includes folders like 'scripts' and 'txt', and files like '.babelrc', '.editorconfig', '.gitignore', 'README.md', 'book.json', 'gulpfile.babel.js', 'package.json', and 'plugins.json'. The 'scripts' folder is highlighted with a red box and the number 4.

Figura 1. Elaboración de los apuntes de una asignatura

La Figura 1 muestra un ejemplo de tarea en la que se colabora en la elaboración de los apuntes de una asignatura. Se trata de un repositorio GitHub en el que participaron 18 estudiantes (1) con nombre 'presentaciones-todos' (3) que pertenece a la organización ULL-ESIT-SYTW-1617 (2), que contiene el conjunto de ficheros que se relacionan (4). Al haber trabajado con GitHub en la elaboración del proyecto se obtienen un conjunto de estadísticas que son las que permiten realizar la evaluación del trabajo realizado por parte de los estudiantes como se muestra en la Figura 2.

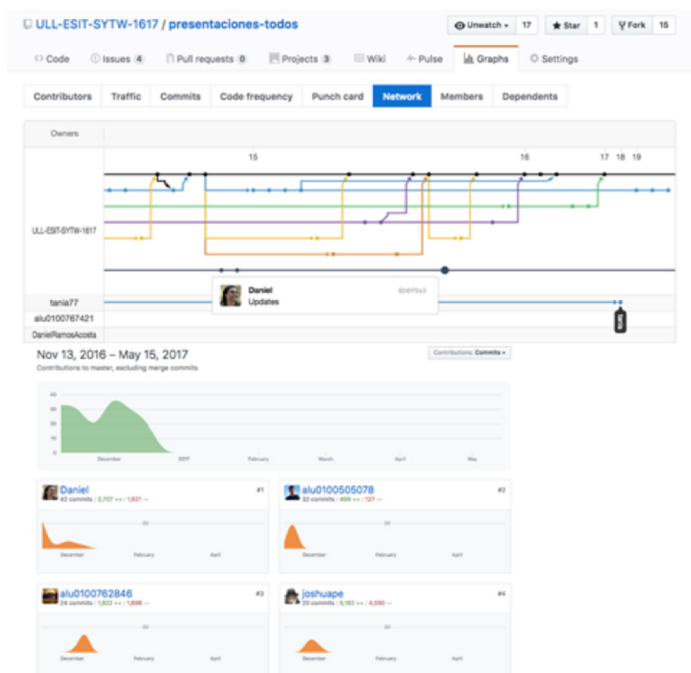


Figura 2. Estadísticas de seguimiento.

A continuación describimos las funcionalidades de GitHub utilizadas en el desarrollo de las asignaturas. Se presentan agrupadas bajo dos criterios:

1. El primero atiende a su uso por parte del profesor como soporte de la asignatura.
2. El segundo se fija en su papel como herramienta de aprendizaje de los alumnos.

## GITHUB CLASSROOM PARA PROFESORES

Para crear y gestionar una asignatura con GitHub Classroom se han de seguir los siguientes pasos:

1. Registrarse en GitHub.
2. Crear una organización para la asignatura.
3. Asociada a cada organización crear un aula en GitHub Classroom.
4. Solicitar a GitHub repositorios privados para la organización.
5. Crear tareas utilizando el aula en GitHub Classroom.
6. Publicar el enlace asociado a la tarea en el Campus Virtual.

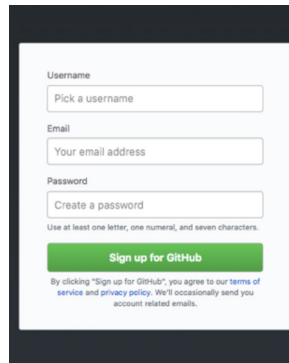
A screenshot of the GitHub registration form. It features four input fields: 'Username' with the placeholder 'Pick a username', 'Email' with 'Your email address', and 'Password' with 'Create a password'. Below the password field is a note: 'Use at least one letter, one numeral, and seven characters.' A green button labeled 'Sign up for GitHub' is positioned below the form. At the bottom, there is a small disclaimer: 'By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account related emails.'

Figura 3. Registro en GitHub.

Para llevar a cabo el primer paso hay que abrir en un navegador la dirección de GitHub (<https://github.com>) y registrarse haciendo uso de la dirección de correo institucional (usuario@ull.edu.es), puesto que después pedirán que se acredite que se forma parte de una institución educativa, véase la Figura 3. El registro se lleva sólo una vez. No es necesario hacerlo cada curso académico.

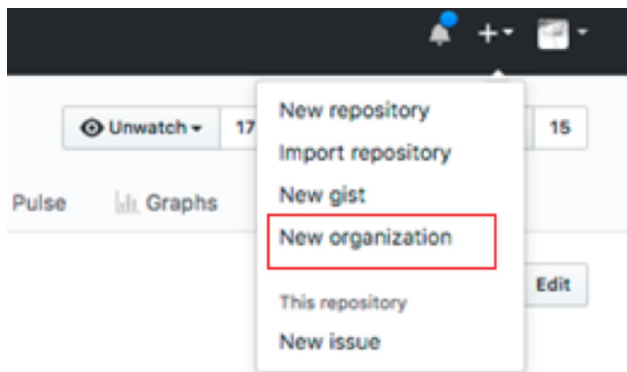


Figura 4. Creación de una organización.

En segundo paso consiste en crear una organización en GitHub (véase la Figura 4). Este paso lo tiene que llevar a cabo cada curso académico el coordinador de la asignatura.

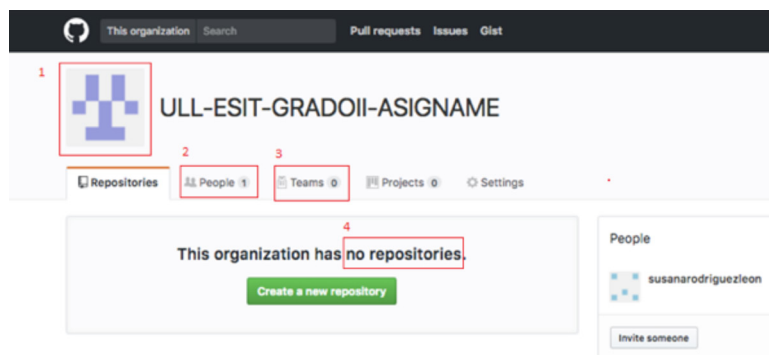


Figura 5. Organización con nombre ULL-ESIT-GRADOII-ASIGNAME.

El resultado de la creación de la organización puede verse en la Figura 5. Se crea un logo por defecto (1) que puede ser modificado por el profesor. En la creación del nombre de la organización se ha utilizado la siguiente nomenclatura: en primer lugar las siglas de la Universidad de La Laguna, a continuación las de la Escuela superior de Ingeniería y Tecnología, en tercer lugar los estudios donde se imparte la asignatura en este caso el Grado en Ingeniería Informática y finalmente el nombre de la asignatura. La única persona que forma parte de la organización es el profesor que la ha creado (2) pero puede añadir más profesores como propietarios o colaboradores de la misma. Tampoco hay creados grupos (3) y la organización se ha creado sin repositorios (4).

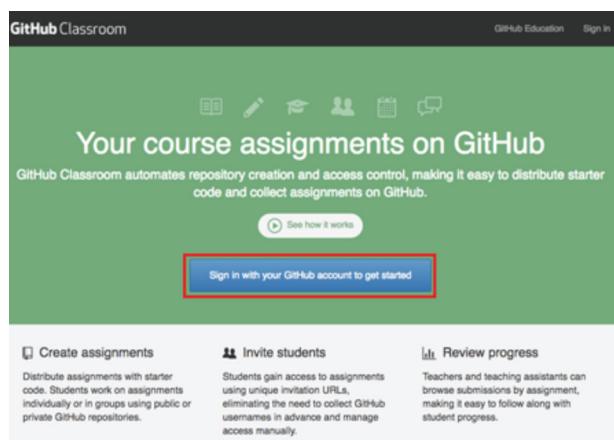


Figura 6. Creación de un aula en GitHub Classroom.



En tercer lugar se ha de crear un aula en GitHub Classroom (<https://classroom.github.com>). Puesto que ya se ha creado una cuenta en GitHub basta con pulsar el botón azul que aparece en la Figura 6. En ese momento se mostrará un aviso indicando que se debe autorizar a Classroom a acceder a la cuenta de GitHub y se pulsa el botón que lo permite. Además, se tiene que indicar a GitHub que le permita a Classroom el acceso a la organización. Finalizadas las autorizaciones aparecerá vinculada al Classroom la organización y se podrá añadir al resto de profesores, así como, crear las tareas (véase la Figura 7).

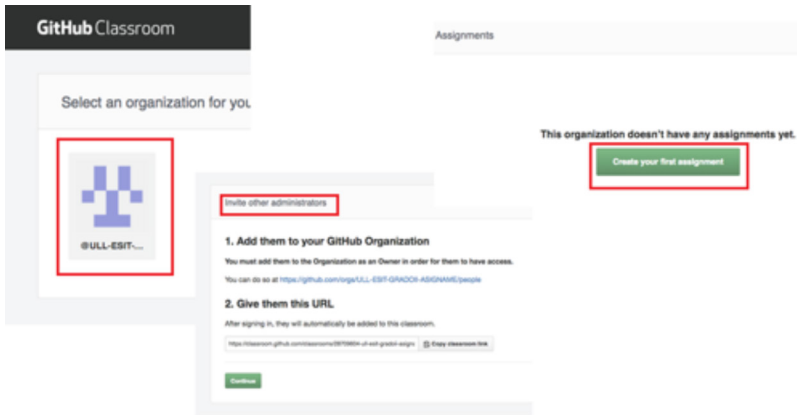


Figura 7. Classroom creada y asociada a la organización

Figura 8. Solicitud de repositorios privados

El paso número 4 consiste en solicitar repositorios privados en <https://education.github.com/>. En el formulario que aparece se ha de especificar el nombre de la organización y del classroom que se han creado en los pasos anteriores, así mismo, es necesario aportar una acreditación de que se es profesor (véase la Figura 8).

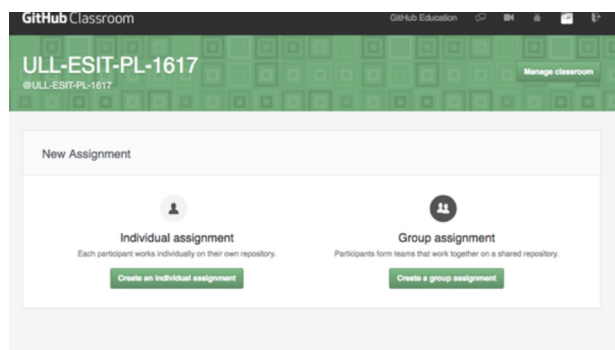


Figura 9. Creación de tareas en GitHub Classroom.

En este momento ya se puede pasar al paso número cinco y crear tareas en GitHub Classroom. La Figura 9 muestra la interfaz de usuario. Las asignaciones pueden ser individuales o en equipo. Si se considera la creación de una tarea en equipo se desplegará el formulario de la Figura 10. El profesor ha de especificar el título de la tarea, elegir entre si el contenido de la misma será público o privado, decidir si los estudiantes tendrán permisos de administrados sobre el repositorio de la tarea y adjuntar el esquema inicial con el que empezarán a trabajar los estudiantes si hay alguno.

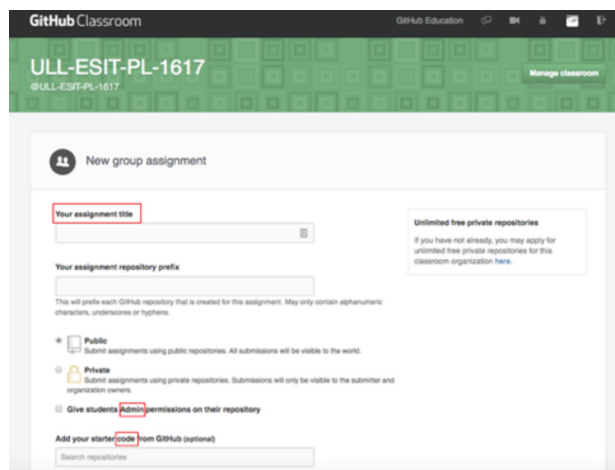


Figura 10. Creación de una tarea en equipo.

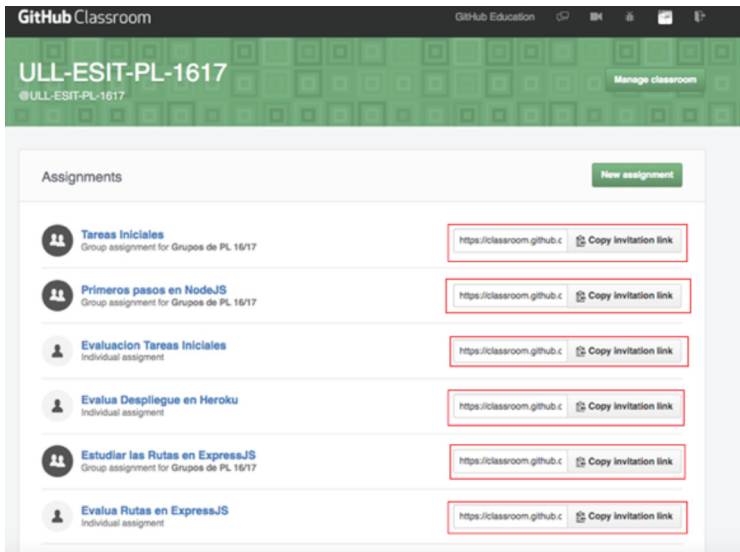


Figura 11. Cada tarea de GitHub Classroom tiene asociado un enlace.

Finalmente, como sexto paso hay que dejar disponible en el campus virtual institucional los enlaces a las tareas que se creen en GitHub Classroom de manera que los estudiantes las puedan aceptar. En la Figura 11 se muestra un ejemplo de tareas con sus enlaces asociados.

## GITHUB CLASSROOM PARA ESTUDIANTES

Los estudiantes tienen que aceptar las asignaciones realizadas por el profesor en GitHub Classroom. Para ello, han de seguir el enlace que el profesor ha dejado en el aula virtual de la asignatura. Tan pronto el alumno pincha en el enlace, se le pide aceptar la asignación (véase la Figura 12). Si es la primera vez que acepta, automáticamente es añadido como miembro a la organización de la asignatura. Si la asignación fue configurada como tarea de equipo y es la primera vez que esto ocurre, el primer alumno crea el equipo y le da nombre y los restantes miembros que lleguen después se unen a ese equipo. Los profesores siempre pueden cambiar los equipos, añadir y/o suprimir miembros de la organización, de los equipos, etc. Cuando se crea una asignación es posible reutilizar un conjunto de equipos previamente creado o crear un conjunto nuevo.

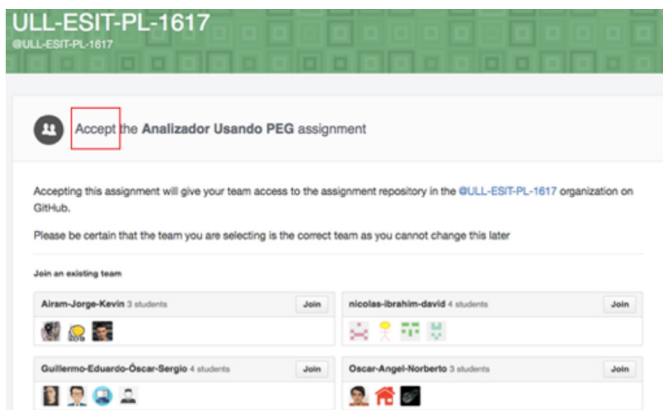


Figura 12. Vista de un alumno cuando pulsa el enlace para aceptar la tarea.

## RESULTADOS Y DISCUSIÓN

Se ha utilizado GitHub Classroom en un conjunto de asignaturas del Grado en Ingeniería Informática de la Universidad de La Laguna. Esta herramienta permite la automatización de la creación de repositorios y el control de acceso a los mismos, lo que ha facilitado la distribución de los distintos ejercicios y la recogida de retroalimentación por parte de los estudiantes.

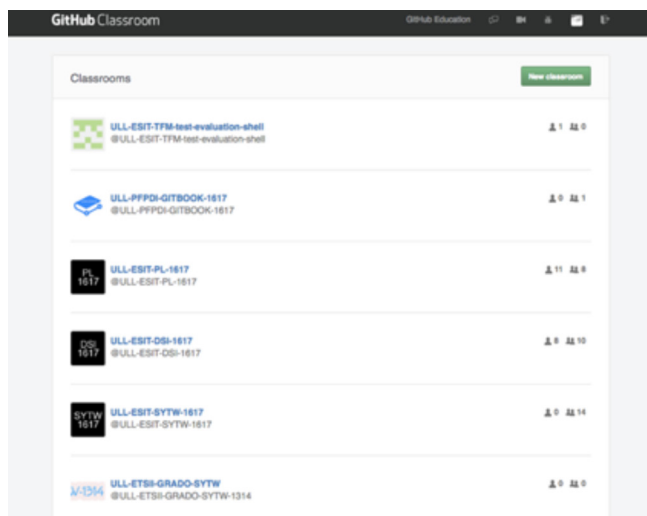


Figura 13. Vista de un profesor del conjunto de asignaturas en GitHub Classroom.

En la Figura 13 se muestra una vista del profesor con distintas asignaturas en GitHub Classroom. En ella se hace coincidir el nombre de la asignatura con el de la organización que se le asocia. Los números que aparecen a la derecha indican la cantidad de tareas individuales y en equipo que contiene el aula. Nótese que un profesor puede tener tantas aulas como necesite. Los alumnos trabajan dentro de la organización. Todos los trabajos del curso están en un solo lugar. Además, se puede controlar quién hizo qué con gran nivel de detalle. En la vista del profesor que se muestra en la Figura 14 todos los proyectos son privados han participado 48 estudiantes y se han creado 28 equipos.

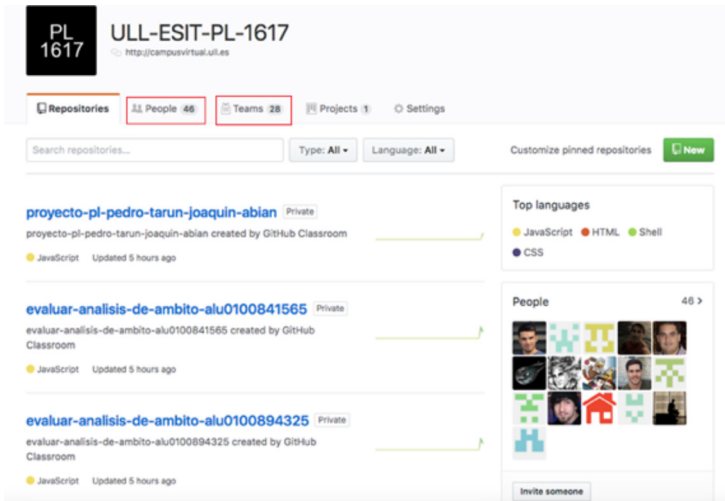


Figura 14. Vista de un profesor de los repositorios de la organización.

## CONCLUSIONES

Se ha descrito e implementado satisfactoriamente un método de aprendizaje colaborativo para varias asignaturas del grado en Ingeniería Informática usando GitHub Classroom como herramienta de gestión. Es necesario profundizar en el análisis de esta plataforma como herramienta docente para dimensionar adecuadamente su papel. Se ha propuesto su integración con otras plataformas educativas como Moodle. Además, esta experiencia ha permitido difundir la formación en las metodologías, técnicas y herramientas asociadas al control de versiones y a los servicios de gestión de repositorios entre la comunidad educativa, lo que fomenta el desarrollo de buenas prácticas para la mejor explotación de este tipo de herramientas.

En trabajos futuros se abordarán aspectos cuantitativos y cualitativos relacionados con la satisfacción de alumnos y profesores, los resultados académicos, y la evolución de la carga docente.

## REFERENCIAS BIBLIOGRÁFICAS

CHACON, S. and STRAUB, B. (2014). *Pro Git*. Berkely, CA, USA: Apress

GitHub Press. (2017). *Git Hub*. 07/05/2018, Sitio web: <https://github.com/about/press>.

GRIFFIN, T. and SEALS, S. (2013). *Github in the classroom: Not just for group projects*. Journal of Computing Sciences in Colleges, 28, 74-74.

LEÓN C., MIRANDA G., RODRÍGUEZ C., SEGREDO E. (2015). *El método científico en la era de los ordenadores*. En Actas de las XXI Jornadas de Enseñanza Universitaria de la Informática (185-192). Andorra la Vella: Universitat Oberta La Salle.

LÓPEZ-PELLICER F., BÉJAR M., LATRE M. (2015). *GitHub como herramienta docente*. En Actas de las XXI Jornadas de Enseñanza Universitaria de la Informática(66-73). Andorra la Vella: Universitat Oberta La Salle.

RODRÍGUEZ C., LEÓN C., MIRANDA G., SEGREDO E., SEGURA C. (2013). *Prácticas de laboratorio en Ruby para Lenguajes y Paradigmas de Programación*. En Innovación Docente en la Educación Superior. Una recopilación de experiencias prácticas aplicadas (547--579). España: Universidad de La Laguna.