

EclipseGavab, un entorno de desarrollo para la docencia online de la programación

Micael Gallego, Francisco Gortázar
Departamento de Ciencias de la Computación
Universidad Rey Juan Carlos
c/ Tulipán s/n, 28933, Madrid
{micael.gallego, francisco.gortazar}@urjc.es

Resumen

En este trabajo se presenta EclipseGavab, un entorno de desarrollo integrado con características colaborativas que permiten la implementación del Aprendizaje Basado en Proyectos en titulaciones online de informática. Dispone, entre otras características, de mensajería instantánea y edición compartida de código, funcionalidades que permiten que el profesor supervise el trabajo de forma telemática y que los alumnos colaboren virtualmente. EclipseGavab soporta la programación en varios lenguajes de programación de distintos paradigmas y características, entre ellos Pascal, C y Java, de forma que pueda utilizarse en diferentes asignaturas a lo largo de la titulación.

1. Introducción

La enseñanza de la informática se está enfrentando a nuevos retos con la irrupción de Internet en la vida cotidiana. Los mecanismos telemáticos son cada vez más utilizados por los estudiantes para comunicarse entre sí, tanto dentro como fuera del ámbito educativo. Aplicaciones de mensajería instantánea como MSN Messenger [16] o Google Talk [14] son uno de los métodos preferidos de comunicación entre los jóvenes. Por otro lado, diversas universidades imparten titulaciones de informática a distancia y utilizan herramientas telemáticas para que los profesores se comuniquen con los alumnos y también para que los alumnos se comuniquen entre sí. Algunas de ellas son la Universidad Nacional de Educación a Distancia (UNED) [19], la Universitat Oberta de Catalunya (UOC) [20] y la Universidad Rey Juan Carlos [11]. Por lo tanto, es cada vez más necesario el desarrollo de herramientas que soporten la enseñanza online de la informática.

Los investigadores están de acuerdo en que el aprendizaje colaborativo en grupo proporciona beneficios pedagógicos [5, 10, 6]. El Aprendizaje Basado en Proyectos o *Project Based Learning* (PBL), es una metodología docente en la que los alumnos adquieren los conocimientos mediante la realización de un proyecto o resolución de un problema adecuadamente diseñado y formulado por el profesor. El Aprendizaje Basado en Proyectos se utiliza ampliamente en la enseñanza presencial de la informática cuando se pide a grupos de alumnos el diseño e implementación de una aplicación. Esta metodología se enmarca dentro del Trabajo Cooperativo Soportado por Ordenador (*computer-supported cooperative work*, CSCW [4, 7]) debido a que el entorno de desarrollo integrado es una aplicación informática que conduce la colaboración en el desarrollo de los proyectos. Debido a que la programación es un proceso complejo, el profesor suele supervisar este proceso de desarrollo. En la enseñanza presencial, esta supervisión habitualmente tiene lugar durante las horas lectivas en aulas con ordenadores, de forma que los alumnos pueden intercambiar opiniones y enfoques acerca del proyecto que tienen que resolver y el profesor puede ayudar con la detección de errores o dando consejos o pistas sobre el desarrollo del mismo.

Actualmente, no existe una implementación satisfactoria del Aprendizaje Basado en Proyectos en la enseñanza online de la informática. Esto se debe fundamentalmente a que los mecanismos de comunicación disponibles en las herramientas colaborativas usadas en este tipo de enseñanza, como Moodle [15] o WebCT [22] no están adaptados para el desarrollo colaborativo de aplicaciones informáticas. Cuando se utilizan este tipo de herramientas de enseñanza online, para

compartir pequeños fragmentos de código fuente entre alumnos o entre el alumno y el profesor, se copia el código desde el entorno de desarrollo y se pega en herramientas de mensajería instantánea o se envía por mail el programa completo. Esto hace que tanto la resolución de dudas por parte de un profesor como la colaboración entre alumnos sean tareas bastante engorrosas y lentas, que se traduce en una merma de calidad comparada con las mismas tareas en la enseñanza presencial.

El uso de Internet como uno de los medios de comunicación preferidos por los estudiantes y la proliferación de las titulaciones online de informática hace que sea de interés disponer de una herramienta de programación colaborativa. Esta herramienta de programación tiene que permitir y facilitar los casos de uso habituales en el contexto de la enseñanza de la programación: colaboración entre alumnos y supervisión del trabajo del alumno por parte del profesor. En este contexto se presenta EclipseGavab, un entorno de desarrollo para la docencia online de la programación.

Este trabajo se estructura de la siguiente forma. En la Sección 2 se realiza una revisión del estado del arte de las herramientas colaborativas para la enseñanza de la programación. En la Sección 3 se describen las características y funcionamiento de EclipseGavab. En la Sección 4 se presentan algunos de los casos de uso colaborativos más habituales en la docencia online de la programación y la forma en la que se llevarían a cabo en EclipseGavab. La validación preliminar de la herramienta y el planteamiento de una experimentación más completa se detallan en la Sección 5. Por último, la Sección 6 muestra las conclusiones y trabajos futuros.

2. Estado del arte

En la literatura científica se pueden encontrar diversos entornos de programación colaborativos. A continuación se describen brevemente algunos de los más relevantes:

- DPE [3]: Esta herramienta toma su nombre de *Distributed Programming Environment* (Entorno de Programación Distribuida). Está formado por un cliente que permite a un usuario, entre otras cosas, la edición

compartida de código, la compilación local y remota y la comunicación a través de *chat* con soporte de audio y vídeo. Por otro lado, dispone de un servidor que permite la administración del sistema, control de versiones de los ficheros fuente, etc. Esta herramienta dispone de mecanismos muy avanzados y completos para el desarrollo colaborativo. No obstante, bajo nuestro punto de vista adolece de un problema importante que reducirá en gran medida su aplicación en un contexto educativo. Esta herramienta no dispone de las funcionalidades de asistencia a la programación presentes en los entornos de desarrollo profesionales como el resaltado de sintaxis, ayuda integrada, compilación incremental, asistentes para la corrección de errores, etc. La falta de estas características hace que los alumnos puedan considerar el uso de otras herramientas más completas aunque dispongan de menos características colaborativas.

- Ripple [1]: Es una herramienta que se integra dentro del entorno de desarrollo profesional Eclipse [13] y permite que ciertas acciones que desarrolla un usuario puedan ser observadas por otro usuario. Las acciones que se propagan de un usuario a otro son el borrado y la creación de ficheros, la edición de código fuente y la ejecución de programas. Además, dispone de una ventana de *chat* que permite la comunicación entre usuarios. Esta herramienta puede utilizarse en diversos casos de uso habituales en la docencia de la programación. Por ejemplo, un alumno puede estar editando un programa y un tutor puede supervisar en tiempo real el proceso. Una de las desventajas de esta herramienta es que está restringida únicamente al lenguaje de programación Java lo que impide su uso en asignaturas que usen otro lenguaje. Otra de las desventajas de esta herramienta es que no permite que el usuario que está observando el desarrollo pueda editar el código fuente. Esto se traduce en que los consejos y propuestas de implementación del observador sólo se pueden llevar a cabo desde la ventana de *chat*, lo que limita su efectividad.

- RECIPE [9]: Es una herramienta formada por un cliente y un servidor que permite el desarrollo colaborativo con las siguientes funcionalidades: Sesión de shell colaborativa, compilación colaborativa, depuración colaborativa y edición colaborativa del mismo fichero de código fuente. Pese a que las características colaborativas son muy completas, bajo nuestro punto de vista adolece del mismo problema que DPE, es decir, no dispone de las facilidades en el desarrollo de código que incorporan los entornos de programación profesionales como Eclipse, lo que hace que los usuarios utilicen otras herramientas.
 - SICAS+PlanEdit+COLLEGE+OOP-Anim [2]: En este trabajo se propone una integración de diversas herramientas educativas colaborativas. Nuestra propuesta no se centra de forma tan específica en la enseñanza de la programación sino más bien en proporcionar mecanismos colaborativos en un entorno de desarrollo de forma que los alumnos puedan colaborar en el desarrollo de aplicaciones. En este sentido, podemos decir que las herramientas educativas colaborativas no entran dentro de nuestro objetivo.
- Con este análisis del estado del arte podemos observar que la principal desventaja de las herramientas colaborativas existentes consiste en que no proporcionan las funcionalidades de las que disponen los entornos de programación profesionales como Eclipse [13], Netbeans [17] o Visual Studio [21]. Entre estas características se encuentran el resaltado de sintaxis, la depuración con entorno gráfico o el marcado de errores de compilación en el código fuente. Por otro lado, algunas de las propuestas anteriores como Ripple [1] sólo permiten la colaboración cuando los programas se desarrollan en el lenguaje de programación Java. Esto limita mucho la utilización de esta herramienta, sobre todo en los primeros cursos de las titulaciones de informática en los que se sigue utilizando mayoritariamente Pascal como primer lenguaje de programación.
- Bajo nuestro punto de vista, un entorno de desarrollo colaborativo debe tener, al menos, las siguientes características:
- *Funcionalidades básicas*: resaltado de sintaxis, errores de compilación marcados en el código, ayuda integrada y depuración con entorno gráfico.
 - *Edición colaborativa*: Esta característica permite a un grupo de programadores ver y editar el mismo fichero de código fuente simultáneamente. Cuando varios programadores trabajan sobre el mismo fichero de código, la edición colaborativa facilita el cambio de rol, pasando fácilmente de ser la persona que edita a la persona que supervisa. Cuando una herramienta colaborativa se utiliza para la supervisión de los alumnos por parte del profesor, la edición colaborativa permite que el profesor añada partes o modifique algunas de las que los alumnos han escrito, mejorando la comunicación.
 - *Varios lenguajes de programación*: La herramienta colaborativa no debe imponer un lenguaje de programación concreto ya que esto limitaría su utilización. Las características colaborativas deben proporcionarse de forma independiente de un lenguaje concreto y la herramienta colaborativa debe estar diseñada de forma que se puedan incorporar nuevos lenguajes de programación cuando sea necesario.
 - *Mecanismo de comunicación integrado*: Se debe incorporar en la herramienta algún mecanismo de comunicación de forma que los usuarios no tengan que utilizar otras aplicaciones o servicios independientes de la misma. Al menos la comunicación debería realizarse mediante *chat*, aunque la comunicación por voz o mediante videoconferencia es deseable.
 - *Envío, recepción y almacenamiento remoto de programas*: Para soportar una efectiva colaboración entre diferentes desarrolladores es necesario que todos puedan gestionar adecuadamente el código desarrollado. Los sistemas de control de versiones como CVS [12] o SVN [18] permiten que el código esté almacenado en un servidor y queden registradas las modificaciones que los desarrolladores realizan sobre el código. Además, este servidor ofrece la oportunidad a un revisor de supervisar la evolución

temporal de los cambios producidos en el código.

Con estas características en mente se ha desarrollado EclipseGavab, una herramienta colaborativa para la docencia online de la programación. En la siguiente sección se describen las características de esta herramienta.

3. Descripción de EclipseGavab

EclipseGavab es una distribución de Eclipse orientada a la docencia presencial y online de la informática. Esta distribución incluye soporte para varios lenguaje de programación, lo que permite su utilización en diferentes asignaturas, y un entorno colaborativo en el cual los alumnos pueden llevar a cabo su aprendizaje. Con EclipseGavab es posible aplicar el Aprendizaje Basado en Proyectos en la enseñanza online. En la Figura 1 se muestra el logotipo de esta herramienta.



Figura 1. Logotipo de EclipseGavab

Como se ha comentado anteriormente, para que una herramienta pueda utilizarse en el contexto de una metodología concreta, es necesario que sea suficientemente flexible. En el caso de la docencia de la programación, en diferentes asignaturas el alumno debe utilizar diferentes lenguajes de programación, y el soporte de éstos debe ser completo. La herramienta EclipseGavab permite el desarrollo de aplicaciones en los lenguajes más habituales en nuestra universidad y en otras universidades. En concreto, los lenguajes de programación soportados son Pascal, Pascal-FC (un dialecto de Pascal para el aprendizaje de la programación concurrente), Java, C y C++, Ruby y Haskell. El

soporte de todos estos lenguajes incluye editores con resaltado de sintaxis y marcación de errores de compilación en el propio código, depuración, ayuda integrada, entre otras características. Todos estos lenguajes se utilizan en diversas asignaturas para enseñar diferentes paradigmas o metodologías de la programación.

Desde el punto de vista colaborativo, EclipseGavab incorpora un conjunto de herramientas que dotan al entorno de desarrollo de capacidades colaborativas como clientes de *chat*, edición compartida del código y un cliente del sistema de control de versiones Subversion [18] que permite utilizar repositorios de código para compartir programas. El control del repositorio subversión se ofrece en EclipseGavab a través del *plugin* Subversive. El *chat* y la edición compartida se implementan con el *plugin* Eclipse Communication Framework. Este *plugin* presenta características similares al módulo Collab de Netbeans.

Las Figura 2 y 3 muestran una sesión de edición compartida de código. La Figura 2 muestra el entorno EclipseGavab que está siendo utilizado por el profesor. La Figura 3 muestra el entorno que está siendo utilizado por el alumno. El profesor incluye un comentario en el código del alumno indicándole que la implementación que ha realizado de una determinada funcionalidad que se le requería en la práctica no considera un caso básico. En su editor, el alumno observa inmediatamente lo que el profesor escribe.

Eclipse es un sistema modular que permite la inclusión de funcionalidades para el desarrollo en diversos lenguajes de programación. Están disponibles componentes para muchos lenguajes de programación, pero al comenzar el trabajo con EclipseGavab no estaba disponible el soporte para Pascal, un lenguaje muy utilizado en los primeros cursos de enseñanza universitaria de la informática. Por este motivo, los autores han desarrollado los módulos necesarios para la incorporación de Pascal como lenguaje de programación de primera clase en Eclipse. Se han desarrollado herramientas de soporte que incluyen edición con resaltado de sintaxis, marcación de los errores de compilación en el propio código, depuración y ayuda integrada.

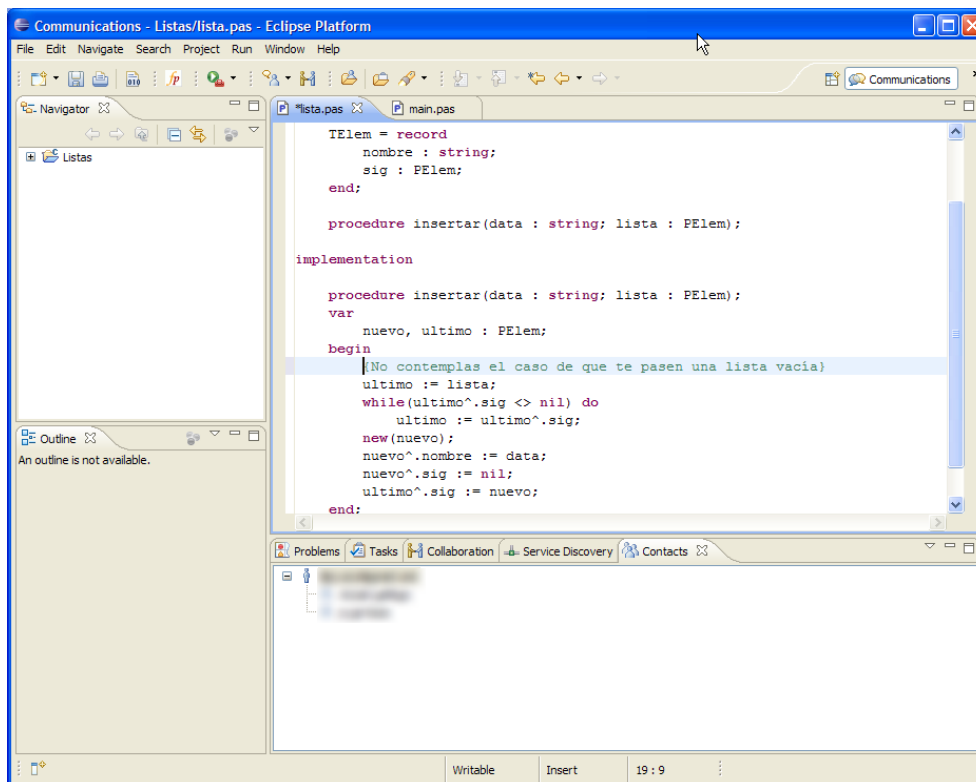


Figura 2. Sesión de EclipseGavab usada por un profesor

EclipseGavab es una herramienta multiplataforma de forma que pueda utilizarse tanto en Windows como Linux. Esto permite que los alumnos no estén obligados a utilizar un sistema operativo concreto para utilizar la herramienta. Además, se ha hecho un esfuerzo para que esta herramienta pueda instalarse con un único instalador, sin necesidad de instalar o configurar ninguna otra aplicación o compilador adicional. Incluye los compiladores y máquinas virtuales necesarios para los lenguajes soportados.

Con la adaptación metodológica necesaria y con la incorporación en un entorno de desarrollo las herramientas colaborativas necesarias pretendemos que el Aprendizaje Basado en Proyectos de la enseñanza online de la informática tenga una calidad similar a la

modalidad presencial, mejorando enormemente la experiencia actual. Además, consideramos que la adaptación metodológica y las conclusiones obtenidas podrán reutilizarse en otras asignaturas que utilizan el Aprendizaje Basado en Proyectos, permitiendo, por ejemplo, la tutorización de prácticas a distancia y el desarrollo colaborativo entre alumnos.

4. Casos de uso colaborativos

En el contexto del aprendizaje basado en proyectos se han identificado diversos casos de uso que requieren de herramientas colaborativas. A continuación se describen algunos de los más habituales y que están soportados por EclipseGavab debido a las diferentes funcionalidades que soporta.

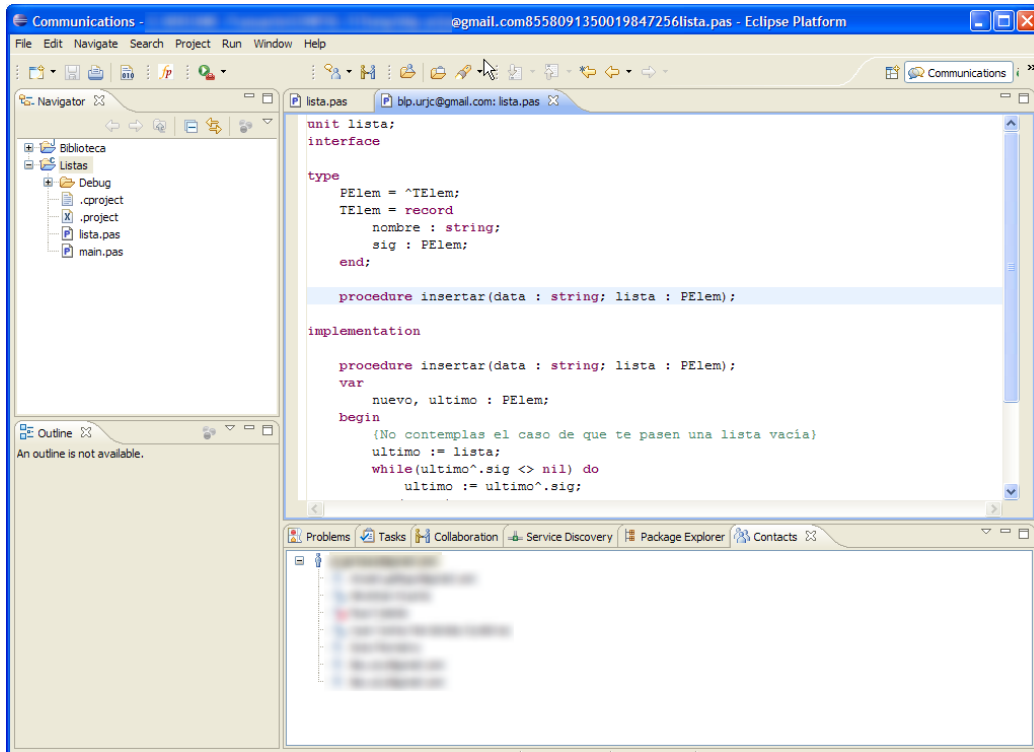


Figura 3. Sesión de EclipseGavab usada por un alumno

4.1. Desarrollo de prácticas en grupo

En este contexto, los alumnos deben realizar una práctica en grupos de dos o más personas. El objetivo de este tipo de prácticas es que desarrollen la capacidad de trabajar en grupo, dividiéndose el trabajo a realizar y después integrándolo. Para ello es necesaria la comunicación entre los miembros del grupo. En este caso, el *chat* es una herramienta fundamental. Si los alumnos disponen de un *chat* integrado en el entorno de desarrollo, pueden tomar decisiones conjuntamente sin que tengan que estar físicamente juntos. Esto permite realizar prácticas en grupos incluso en asignaturas online. Compartir el código usando un repositorio de código remoto elimina la necesidad de enviar el código usando el correo electrónico, facilitando enormemente el proceso. Además, los cambios producidos por cada

integrante del grupo son registrados en el servidor. Por último, el repositorio puede ser accesible para el profesor, quien se lo proporciona a los alumnos, y lo puede utilizar para tutorizar el desarrollo de las prácticas. Por ejemplo, el profesor podría pasar detectores de errores y de malas prácticas de manera automática (todas las semanas) revisando en mayor detalle aquellos proyectos que presenten mayores deficiencias en este sentido, dado que son los que corren mayor riesgo de sufrir retrasos y problemas en el momento de la entrega.

4.2. Clases prácticas

En este contexto el profesor puede realizar un seguimiento de los progresos de un alumno, e incluso utilizando la edición compartida, ayudarle a superar alguno de los obstáculos iniciales. El profesor puede publicar una primera

versión (un bosquejo, por ejemplo) de la solución a un determinado problema en un repositorio de código remoto y los alumnos se lo pueden descargar desde el mismo entorno. Los alumnos pueden entregar sus prácticas utilizando el propio repositorio, y el profesor gestiona la fecha límite de entrega suspendiendo el acceso al repositorio de todas las cuentas en el momento en que dicha fecha vence.

4.3. Tutorización de alumnos

Determinadas herramientas colaborativas pueden mejorar la experiencia docente. En este caso de uso, un alumno con una duda relativa a una práctica o un ejercicio que está implementando, necesita ayuda del tutor. Lo habitual en este tipo de casos es que si la asignatura es presencial, el alumno vaya a ver físicamente al tutor con el programa en algún dispositivo electrónico para consultarlo; si la asignatura es online, el alumno y el tutor se pueden intercambiar el programa por correo electrónico. Ahora bien, si alumno y profesor cuentan con un entorno de desarrollo colaborativo que permita edición compartida del código, entonces ambos pueden ver de manera simultánea el código del alumno. El tutor puede hacer sugerencias a través del *chat*, e incluso modificar de manera remota el código del alumno. Las modificaciones pueden incluir la inserción de comentarios sobre cosas que deberían cambiarse o problemas que pueden presentarse debidos a ciertas decisiones tomadas en el código.

5. Validación de la herramienta

La herramienta propuesta ha sido utilizada los dos últimos años en la enseñanza de las estructuras de datos, la programación funcional, la programación orientada a objetos, la programación concurrente y en una asignatura de aprendizaje de los lenguajes C y C++. En total, más de 300 alumnos la han utilizado cada año.

Hasta el momento, la forma de evaluación ha consistido en la realización de una encuesta a los alumnos sobre el grado de satisfacción con la herramienta. En los dos años en los que se realizó esta encuesta, los alumnos demostraron

una clara preferencia por EclipseGavab, frente a las herramientas que habían utilizado anteriormente como Turbo Pascal.

Para el curso académico 2008/2009 proponemos una encuesta de evaluación mucho más concreta que nos permita determinar de forma efectiva la aportación a los alumnos de los mecanismos colaborativos. Se realizará una encuesta previa a la utilización de la herramienta, que muestre la opinión de los alumnos en su experiencia en el desarrollo colaborativo sin haber usado la herramienta. Después del uso de la herramienta se les volverá a hacer otra encuesta para determinar en qué medida el uso de la herramienta ha mejorado su experiencia en la colaboración entre alumnos y en la comunicación con el profesor. También se les pedirá que indiquen las mejoras que consideren tanto en la herramienta como en la metodología docente que en la que se utiliza.

6. Conclusiones y trabajos futuros

En este trabajo se presenta EclipseGavab, una herramienta colaborativa para la enseñanza online de la programación. Se han revisado otras herramientas colaborativas del estado del arte y se han identificado las funcionalidades esenciales en este tipo de herramientas. Se han descrito diversos casos de uso colaborativos en la enseñanza online que se pueden llevar a cabo con EclipseGavab. Por último se indica cómo la herramienta ha sido utilizada con éxito en los dos últimos años y se plantean encuestas más elaboradas para determinar con precisión el impacto de la herramienta en el trabajo colaborativo de los alumnos.

EclipseGavab está compuesto por varias herramientas y plugins con las licencias libres GPL y EPL (entre otras). Se puede descargar de la web <http://www.gavab.es/eclipsegavab>. Es importante mencionar que EclipseGavab es una distribución de Eclipse completa, que incluye todo lo necesario (*plugins* de Eclipse y compiladores) para el desarrollo de aplicaciones. No obstante, los *plugins* de los que se compone se pueden instalar individualmente en cualquier distribución de Eclipse.

Como trabajos futuros se plantea incorporar capacidades multimedia de comunicación, como audio y vídeo, para facilitar la colaboración.

También se pretende incluir más servicios compartidos como son: pizarra virtual, presentaciones, depuración compartida, etc.

Agradecimientos

Durante el desarrollo EclipseGavab muchos profesores y alumnos han participado evaluando y probando la herramienta y aportando ideas sobre nuevas funcionalidades. Parte del desarrollo de EclipseGavab se ha llevado a cabo en el marco de un Proyecto de Innovación Educativa dentro de la IV Convocatoria de Ayudas a la Innovación Educativa para el curso 2008/2009 de la Universidad Rey Juan Carlos.

Referencias

- [1] Boyer, K. E., August A. Dwight, R. Taylor Fondren, Mladen A. Vouk, and James C. Lester. "A Development Environment for Distributed Synchronous Collaborative Programming". In *Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE '08)*, Madrid, Spain, 2008, 158-162.
- [2] Bravo, C., Marcelino, M.J., Gomes, A., Esteves, M., Mendes, A.J. "Integrating educational tools for collaborative Computer Programming learning", *Journal of Universal Computer Science*, Volumen 11(9), 1505-1517, 2005.
- [3] Chang-Hyun, J. and Allen J. Arnold, "A Portable and Collaborative Distributed Programming Environment", *The 2003 International Multi-Conference in Computer Science and Computer Engineering – The International Conference on Software Engineering, (IMCCSCE – SERP'03)*, 198-203, Las Vegas, Nevada, June 23-26, 2003.
- [4] Kouzes, R. T., J. D. Myers, and W. A. Wulf, "Doing Science on the Internet", *IEEE Computer*, Volumen 29(8), 40-46, 1996.
- [5] Livingstone, D. and Lynch, K. "Group Project Work and Student-centred Active Learning: two different experiences", *Studies in Higher Education*, Volumen 25(3), 325-345, 2000.
- [6] Morrison, J. "Facilitating Collaborative Learning Within Programming Projects", *Issues in Information Systems*, Volumen 5(1), 2004.
- [7] Palmer, J. D. and N. A. Fields, "Computer-Supported Cooperative Work: Guest Editors' Introduction", *IEEE Computer*, 15-16, 1994.
- [8] Shen, H. "Internet-Based Collaborative Programming Techniques and Environments", Tesis Doctoral, Griffith University, 2003.
- [9] Shen, H. and C. Sun. "RECIPE: A Web-based Environment for Supporting Real-time Collaborative Programming", In *Proceedings of International Conference on Networks, Parallel and Distributed Processing*, pages 283–288, Tsukuba, Japan, ACTA Press, 2002.
- [10] Slavin, R.E. *Cooperative Learning: Theory, Research and Practice*, NJ: Prentice Hall, 1990.
- [11] Campus Virtual de la Universidad Rey Juan Carlos (URJC): <https://www.campusvirtual.urjc.es/>
- [12] CVS: <http://ximbiot.com/cvs/wiki/>
- [13] Eclipse: <http://www.eclipse.org>
- [14] Google Talk: <http://www.google.com/talk/>
- [15] Moodle: <http://www.moodle.org/>
- [16] MSN Messenger: <http://messenger.live.com/>
- [17] NetBeans: <http://www.netbeans.org/>
- [18] Subversion (SVN): <http://subversion.tigris.org/>
- [19] Universidad Nacional de Educación a Distancia (UNED): <http://www.uned.es>
- [20] Universitat Oberta de Catalunya (UOC): <http://www.uoc.edu/>
- [21] Visual Studio: <http://microsoft.com/VisualStudio/>
- [22] WebCT: <http://www.webct.com/>