



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

# **Aprendizaje colaborativo en programación de computadores**

**Jhon A. Méndez**

Universidad Nacional de Colombia

**PLaS Research Group**

***Programming Languages and Systems***

Contact: [jmendezl@unal.edu.co](mailto:jmendezl@unal.edu.co)

# Contenido

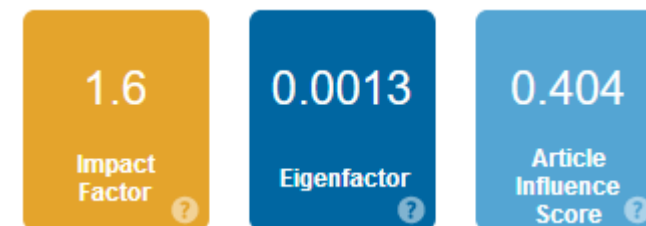
1. Introducción
2. Presentación artículo
3. Referencias

# Introducción

- Los Scripts **CSC**L son los elementos de **diseño** más importantes en la construcción de herramientas para el aprendizaje colaborativo.
- Describe la **forma** o **prácticas** en que los estudiantes tienen que colaborar: distribución de tareas o roles, reglas de turnos, fases de trabajo, entregables, etc.  
(Dillenbourg and Jermann, 2007).
- Los **Scripts** de CSCL se describen como una secuencia de fases cuyos **atributos** contemplan:
  - Tipo de la tarea
  - Formación de grupos
  - Distribución de la tarea
  - Tipo y modo de interacción (remota vs local, síncrona vs asíncrona, etc)
  - Tiempo(Dillenbourg, 2002).

# Presentación artículo

- **NOMBRE:** Evaluating the Effects of Scripted Distributed Pair Programming on Student Performance and Participation.
- **REVISTA:** IEEE TRANSACTIONS ON EDUCATION
- **AÑO PUBLICACIÓN:** VOL. 59, NO. 1, FEBRUARY 2016
- **CONTENIDO:**
  - INTRODUCTION
  - SCEPPSYS SYSTEM
  - SCRIPT
  - EXPERIMENTAL DESIGN
  - EVALUATION RESULTS
  - DISCUSSION
  - CONCLUSION



<https://ieeexplore.ieee.org/document/7089313>

[Despina Tsompanoudi](#)

[Maya Satratzemi](#)

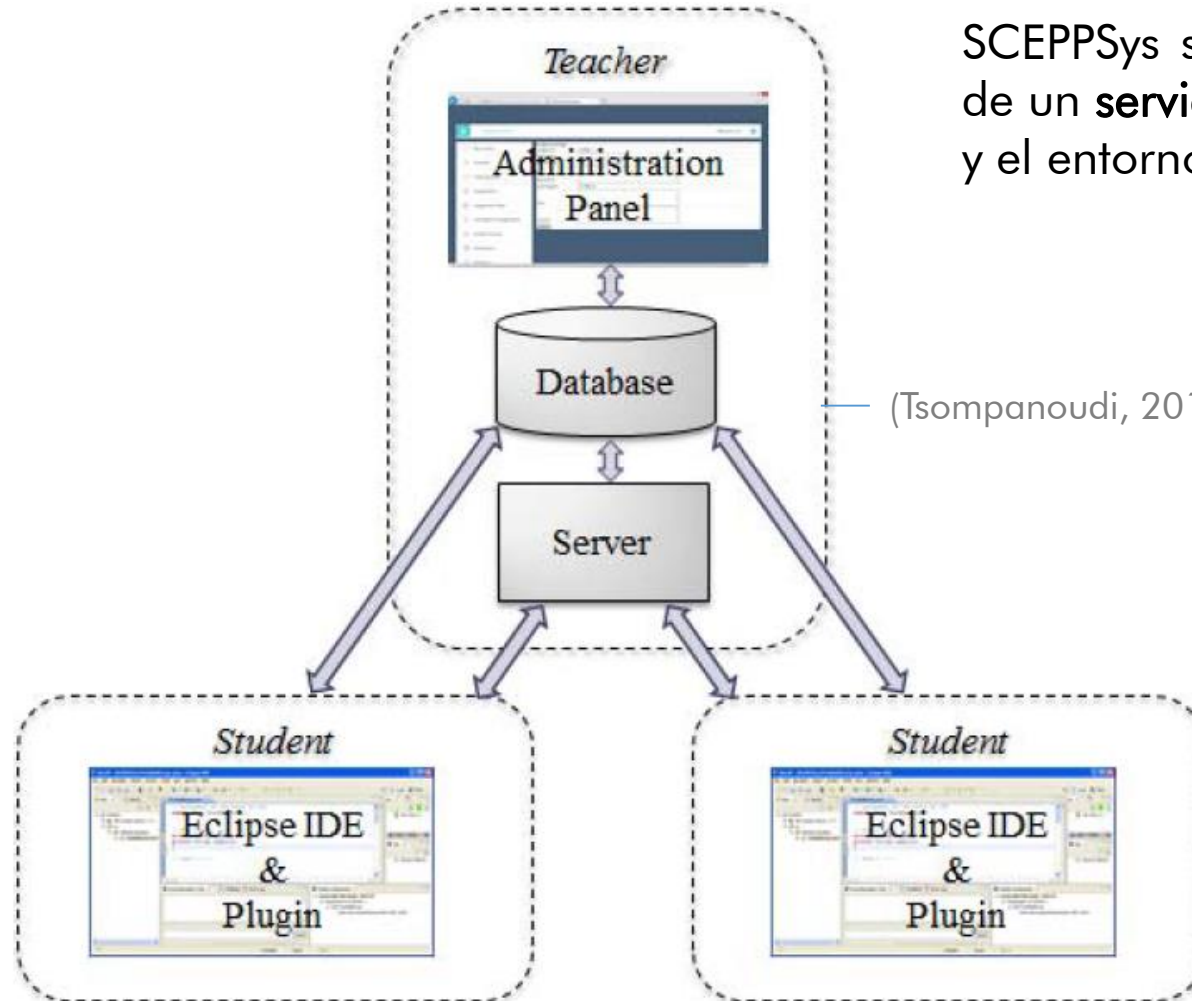
[Stelios Xinogalos](#)

Dept. of Applied Informatics, University of  
Macedonia,

# Presentación artículo: Introducción

- **Áreas de investigación:** programación de pares distribuidos (**DPP**), aprendizaje colaborativo asistido por computadora (**CSCL**, por sus siglas en inglés) y programación de pares (**PP**).
- Existen un número considerable de soluciones DPP disponibles para el IDE de Eclipse, sin embargo carecen de **apoyo educativo**.
- El estudio incluye el diseño de la Guion colaborativo en sistema educativo de programación por pares (**SCEPPSys**), un sistema más orientado a la educación para el uso de DPP en el aula.
  - **Características Plugin:** guarda y analiza las interacciones de los usuarios y ayuda a los educadores a organizar y monitorear las clases de DPP como también la evaluación.
- Comparan **SCEPPSys** con **Sangam** [7], **RIPPLE** [8] y **XPairtise** [9].
- Los **guiones de colaboración** se usan ampliamente en actividades de aprendizaje colaborativo porque proporcionan una manera de estructurar las **interacciones** de los alumnos.

# Presentación artículo: Sistema SCEPPSYS



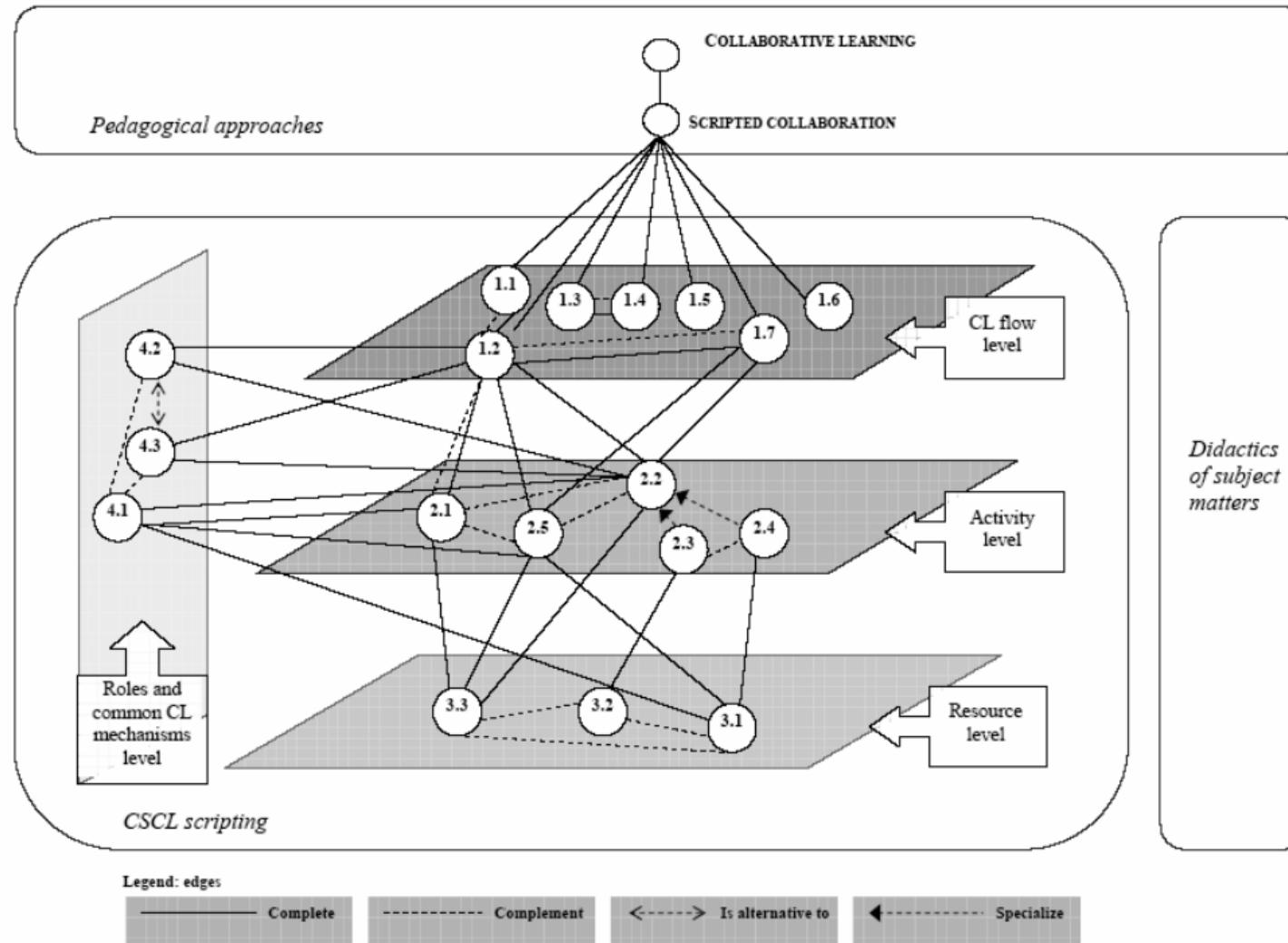
SCEPPSys se basa en una arquitectura **cliente-servidor** y consta de un **servidor**, una **base de datos**, un complemento para **Eclipse** y el entorno de **administración basado en la web**.

(Tsompanoudi, 2014).

(Tsompanoudi, 2016).



Basado en XecliP + Guiones colaborativos


# Presentación artículo: Sistema SCEPPSYS





(Hernandez, 2010).


# Presentación artículo: Sistema SCEPPSYS


 AssignmentSteps Welcome, despina 


 My courses


 Students


 Learning Goals


 Assignments

 Assignment Steps

 Timetable Of Assignments

 Student Groups

 Assessment

 Statistics

AssignmentSteps

Assignment: 3: Project1 ▼

Class name: Product

Step number: 1

Description:

The goal of this assignment is to create an electronic shopping cart which can maintain a number of products. Open the class named Product and declare three fields in order to store the name, the price and the weight of the product, and a fourth field which indicates if the product is on offer or not.

Learning goal: 1: Declare Java primitive data types ▼

Hint:

Declare two private fields of type double, one private field of type String and one private field of type boolean

Solution:

```
private String name;
private double price;
private double weight;
private boolean offering;
```

Submit



# Presentación artículo: Sistema SCEPPSYS(administración)

1. **Participantes:** profesores y alumnos que participan en la colaboración.
2. **Roles:** programador y evaluador (PP)
3. **Grupos:** fijos.
4. **Formación grupos:** determina los resultados de la colaboración.
  1. **Selección libre:** elección según las preferencias de los estudiantes o del profesor.
  2. **Grupos aleatorios:** permite la creación de grupos heterogéneos.
  3. **Niveles de habilidad comparables:** la investigación sugiere que emparejar estudiantes con niveles de habilidad similares tiene resultados positivos sobre motivación y participación [17], [18].
  4. **Niveles de contribución comparables:** esta política también Depende de datos de asignaciones de programación pasadas.

# Presentación artículo: Sistema SCEPPSYS(administración)

- 5. **Distribución tarea:** distribución de las tareas con respecto a los roles (PP).
  - 5. **Colaboración libre:** controlados por el estudiante.
  - 6. **Roles rotativos:** controlados por el sistema.
  - 7. **Adquisición de conocimientos equilibrados:** controlado por el sistema
- 6. **Actividades:** Para cada paso de la tarea el maestro debe proporcionar una descripción del problema y una sugerencia para la solución.
- 7. **Secuenciación:** orden en que se distribuyen tareas y roles.

# Presentación artículo: Sistema SCEPPSYS(administración)

## Evaluación y análisis.

- **Sumativa, por medio de un formulario de evaluación:** el sistema supervisa los **interacciones** con el fin de realizar un seguimiento de la colaboración
- Con el fin de evaluar la contribución de cada estudiante a la tarea asignada, el sistema guarda los siguientes valores:

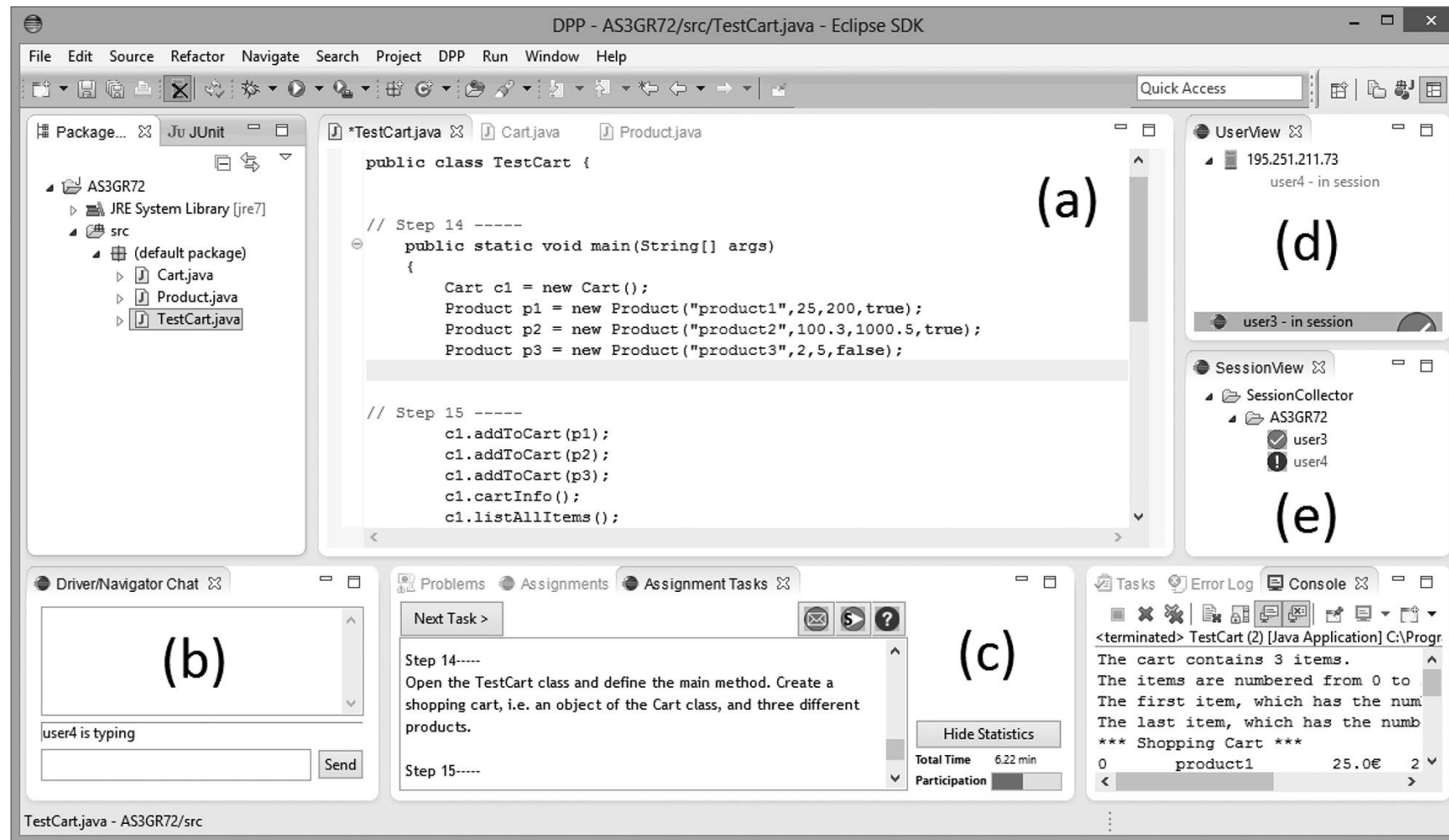
- Tiempo de **programación** de cada alumno.
- Tiempo de **evaluación** de cada alumno.
- Cantidad de código de programa insertado por alumno.

### Genéricos

- Número de **mensajes** intercambiados (chat).
- Número de **ejecuciones** del programa.
- Número de **cambios** de rol.
- Número de **pistas** recuperadas.
- **Distribución** de tareas por alumno.
- **Productividad**: la tasa de programación de cada grupo.
- **Calificación**: promedio de las demás tareas

### Colaboración

# Presentación artículo: Sistema SCEPPSYS(cliente)



- (a) editor compartido.
- (b) la herramienta de chat incrustada.
- (c) las instrucciones del script para la declaración de problema de tarea.
- (d) indicadores de roles conciencia del estado del usuario.
- (e) asignación de roles.

(Tsompanoudi, 2016).

# Presentación artículo: SCRIPT

- La formación de grupos se basó en las calificaciones de los estudiantes en **cursos anteriores**.
- Durante las sesiones de DPP, a los estudiantes se les asignaron **automáticamente los roles**.
- En la **mitad de los grupos**, el sistema realizó sucesivas **alternancias de roles**.
- Para los **otros grupos**, la política de distribución de tareas se basó en el objetivo de aprendizaje de la actividad y **las habilidades individuales** de los estudiantes.
- Las **habilidades individuales** de los estudiantes para cada objetivo de aprendizaje se estimaron a partir de sus calificaciones en **actividades anteriores** que se habían evaluado con la ayuda del sistema.
- Para cada **actividad individual**, el mecanismo de distribución de tareas asignó el rol de **conductor** al estudiante que tenía menos experiencia en el área de ese objetivo.
- Todos los grupos de estudiantes resolvieron las mismas tareas de programación de Java

(Tsompanoudi, 2016).

# Presentación artículo: Experimento

- **Universidad:** Macedonia
- **Asignatura:** Diseño Orientado a Objetos y Programación.
- **Características:** 1h magistral y 2h laboratorio, 13 semanas duración
- **Calificación:** 20%. *resolución de **ocho** proyectos de Java, 10% exámenes intermedios, 10% participación en tareas de laboratorio.*

En 2013, las ocho asignaciones del proyecto había que resolverlo en **parejas** en lugar de individualmente. Los principales objetivos de este estudio fueron investigar el **impacto de DPP** en auténticas condiciones de aprendizaje, y para probar la eficiencia de dos estrategias de distribución de tareas diferentes.

Participaron setenta y cuatro estudiantes (**28 mujeres y 46 hombres**).

(Tsompanoudi, 2016).

# Presentación artículo: Conclusiones

- **Tasa de aprobación:** La tasa de aprobación en 2014 es significativamente superior al de años anteriores en un 69%, **con respecto a años anteriores**.
- **Rendimiento:** la condición experimental obligó a los estudiantes a participar con la mayoría de los conceptos de Java y, en consecuencia para realizar mejor los **exámenes**.
- **Tasa de colaboración:**  
existe una **limitación** del diseño actual ya que captura solo los **caracteres del código fuente** escritos por el programador y no aquellos escrito por el evaluador (como resaltando errores en el código o comentando en el chat mientras el controlador escribe el código).  
  
El sistema **no admite el análisis de contenido**, y esto también es difícil Para lograrlo, dado que los alumnos también utilizan comunicación externa. (solo el 19% utilizaron el chat).

(Tsompanoudi, 2016).

# Presentación artículo: Conclusiones

- **Distribución de tareas:** Las tareas se resolvieron en aproximadamente el **mismo tiempo**.
- **Formación de grupos:** Alrededor del **84%** de los estudiantes declararon en la encuesta final que estaban satisfechos con su pareja, y aunque tenían la oportunidad de cambiar de pareja después de los exámenes parciales, **eligieron quedarse con el mismo**.
- **Realimentación de un docente:** las preguntas fueron más avanzadas de lo habitual, se redujo el tiempo de realización de las tareas y un aumento en la participación de los estudiantes.

(Tsompanoudi, 2016).



# Referencias

- (Tsompanoudi & Satratzemi, 2014; Tsompanoudi, Satratzemi, & Xinogalos, 2016) Tsompanoudi, D., & Satratzemi, M. (2014). A web-based authoring tool for scripting distributed pair programming. *Proceedings - IEEE 14th International Conference on Advanced Learning Technologies, ICALT 2014*, (Section IV), 259–263. <https://doi.org/10.1109/ICALT.2014.81>
- Tsompanoudi, D., Satratzemi, M., & Xinogalos, S. (2016). Evaluating the Effects of Scripted Distributed Pair Programming on Student Performance and Participation. *IEEE Transactions on Education*, 59(1), 24–31. <https://doi.org/10.1109/TE.2015.2419192>
- [17] L. Williams, L. Layman, J. Osborne, N. Katira, and N. Carolina, “Examining the Compatibility of Student Pair Programmers,” *Agile Conference*, 2006.
- [18] N. Zacharis, “Evaluating the Effects of Virtual Pair Programming on Students’ Achievement and Satisfaction,” *International Journal of Emerging Technologies in Learning (iJET)*, vol. 4, no. 3, pp. 34–39, Sep. 2009.

(Tsompanoudi, 2016).

# Gracias!

**PLaS Research Group**  
***Programming Languages and Systems***  
Contact: [jmendezl@unal.edu.co](mailto:jmendezl@unal.edu.co)



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA