

## Creación de grupos de trabajo con afinidad de perfiles para una Aplicación Móvil tipo E-Learning

M. Palacios Hernández <sup>1</sup>, L. A. Reyes Hernández <sup>1</sup>, A. López Chau <sup>2</sup>, I. López Martínez <sup>1</sup>, U. Juárez Martínez <sup>1</sup>

<sup>1</sup>División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Orizaba, Orizaba, Veracruz México,

<sup>2</sup> Universidad Autónoma del Estado de México, Zumpango Estado de México.

\*luzpalacios1511@gmail.com

Área de participación: Investigación Educativa

### Resumen

Hoy en día el E-Learning ha marcado un cambio en el área educativa, ya que ha beneficiado a los estudiantes que no les es posible estudiar de manera presencial en un aula. Existen aplicaciones E-Learning que ayudan a los estudiantes a tomar sus clases, aunque existe el inconveniente de que los estudiantes no se hallan presencialmente en un mismo lugar siendo un inconveniente para la formación de grupos de trabajo. En este artículo se presenta una arquitectura en la que se muestra el módulo de recomendación para la creación de grupos de trabajo con perfiles afines, usando los tweets extraídos de cada usuario de Twitter para la Aplicación Móvil tipo E-Learning Mochuelo Learning utilizando algoritmos de agrupamiento para la formación de grupos de trabajo.

**Palabras clave:** Aplicación Móvil, Twitter, E-Learning.

### Abstract

Nowadays E-Learning has marked a change in the educational area, since it has benefited students who cannot study in person in a classroom. There are E-Learning applications that help students to take their classes, although there is the disadvantage that students are not in the same place in one place being an inconvenience for the formation of work groups. In the present this article presents an architecture in which the recommendation module for the creation of groups of and work with related profiles is shown, using the tweets extracted from each Twitter user for the Mochuelo Learning E-Learning Mobile Application using clustering algorithms for the formation of work groups.

**Key words:** Mobile Application, Twitter, E-Learning.

### Introducción

El desarrollo de aplicaciones ha beneficiado no solo en el entretenimiento, sino, también en el área educativa con la aparición del E-Learning el cual favorece a los alumnos que no pueden tomar clases de manera presencial dentro de un aula. Por otro lado, las redes sociales cada día están presentes en los usuarios de cualquier edad, siendo los jóvenes los mayores involucrados.

Por tal motivo se propone una arquitectura basada en el patrón arquitectónico Modelo-Vista-Controlador, dividiendo en tres componentes principales: Vistas, Modelo y Controlador ayudando a una fácil relación de los componentes.

La arquitectura cuenta con el módulo de recomendación para la creación de grupos de y trabajo con perfiles afines, usando los tweets extraídos de cada usuario de Twitter para la Aplicación Móvil tipo E-Learning Mochuelo Learning [1] utilizando algoritmos de agrupamiento para la formación de grupos de trabajo. La arquitectura cuenta con el módulo de recomendación para la creación de grupos de trabajo con perfiles afines usando la biblioteca Twitter4j que proporciona Twitter para extraer los tweets de cada usuario y a su vez desempeñándose como

medio de autenticación para acceder a la aplicación móvil tipo E-Learning Mochuelo Learning ya desarrollada y formar grupos de trabajo.

## Metodología

Como metodología de desarrollo se empleará la metodología Mobile-D la cual es usada para el desarrollo de aplicaciones móviles, parte como creación del proyecto "ICARUS" en el 2004, posee cualidades de muchas otras metodologías como ser eXtreme Programming, Crystal Methodologies y Rational Unified Process [2].

Las ventajas de la metodología son las siguientes:

- Un costo bajo al realizar un cambio en el proyecto.
- Entrega resultados de manera rápida.
- Asegura el software adecuado en el momento adecuado.

La metodología también cuenta con las siguientes desventajas:

- No sirve para grupos de desarrollos grandes y segmentados.
- Depende de buena comunicación entre los miembros del equipo.

Mobile-D tiene el objetivo de ser una metodología de resultados rápidos, con mira a grupos de pocas personas o pequeños grupos, los integrantes del grupo deben poseer una habilidad y capacidad similar entre todos [2].

## Fases

La metodología cuenta con 5 fases por las cuales pasa el producto para realizarse:

### Fase de exploración

Se centra la atención a la planificación y a los conceptos básicos del proyecto. Se realizan los alcances del proyecto y su establecimiento con las funcionalidades donde se va a llegar.

El propósito de esta fase es la planificación y establecimiento de una buena planificación, esta fase es muy importante para establecer las bases para una implementación bien controlada de software, la arquitectura del producto, el proceso de desarrollo y la selección del medio ambiente [3].

### Fase de inicialización

En la iniciación se configura el proyecto y se preparan todos los recursos necesarios, se le dedica un día a la planificación y el resto al trabajo y publicación.

El propósito de esta fase es permitir el éxito de las siguientes fases del proyecto mediante la preparación y verificación de todas las cuestiones fundamentales del desarrollo a fin de que todos están en plena disposición de la aplicación de los requisitos seleccionados por el cliente [3].

### Fase de producto

Antes de iniciar el desarrollo de una funcionalidad debe existir una prueba que verifique su funcionamiento, en esta fase se lleva a cabo toda la implementación de los módulos.

El propósito en la fase de producción es implementar la funcionalidad requerida en el producto mediante la aplicación del ciclo de desarrollo iterativo e incremental [3].

### Fase de estabilización

En esta fase se llega la integración para vincular los módulos separados en una única aplicación. El propósito de la fase de estabilización es asegurar la calidad de la implementación del proyecto [3].

### Fase de pruebas

Se pasa al testeo hasta tener una versión estable del producto según lo establecido por el cliente. Si es necesario se reparan errores, pero no se desarrolla nada nuevo. Una vez terminado todas las fases se debería contar con una aplicación publicable y entregable al cliente.

El propósito de la fase de pruebas es ver si el sistema productor implementa la funcionalidad definida del cliente correctamente, proporcionar la retroalimentación al equipo de desarrollo de los defectos y errores encontrados en la funcionalidad del software para ser corregidos estos defectos encontrados [3].

### Arquitectura propuesta

Para el desarrollo del Módulo Recomendación para la Creación de Grupos de Trabajo con Perfiles Afines se tomó como cimiento la arquitectura de la Aplicación Móvil tipo E-Learning incorporando el modulo antes mencionado. Dicha arquitectura se basa en el patrón de diseño Modelo-Vista-Controlador (MVC). Este patrón [4] separa la aplicación en tres partes: la vista es la representación visual de los datos, todo lo que tenga que ver con la interfaz gráfica; el controlador es el encargado de recibir las órdenes del usuario y solicitar los datos al modelo y comunicárselos a la vista; el modelo administra la conducta y los datos de la aplicación y responde a las instrucciones (usualmente del controlador) para cambiar el estado. En la figura 1 se muestra la arquitectura a utilizar para el desarrollo del Módulo de Agrupamiento de Perfiles.

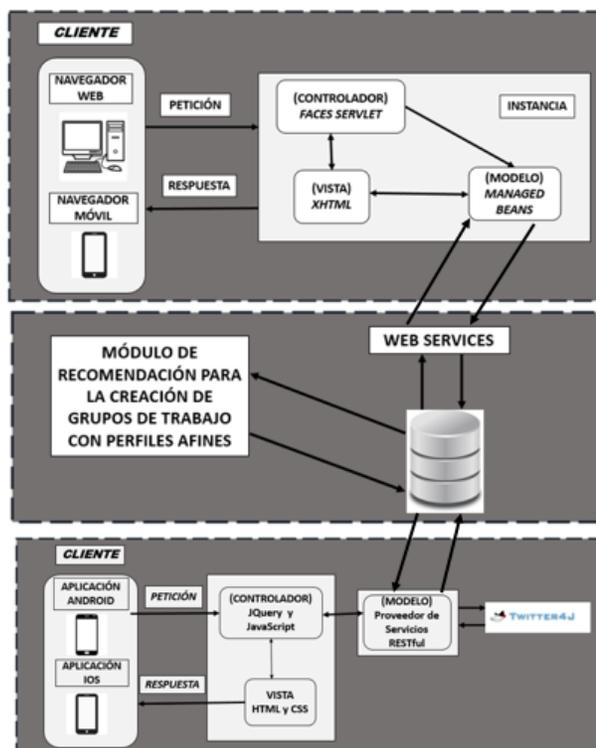


Figura 1. Arquitectura propuesta.

### Arquitectura del apartado Web

El apartado Web se encuentra desarrollado e implementado, dividiéndose de la siguiente manera:

**Cliente:** es el usuario que solicita el servicio a través de un navegador Web, haciendo la petición al Controlador requiriendo la información que necesita.

**Controlador:** es el componente Faces Servlet el cual recibe las peticiones del Cliente, así mismo encargándose de ser el intermediario entre el modelo y la vista.

**Modelo:** unen la lógica de la aplicación representada en los beans administrados, los cuales pasan información a los beans de modelo. Estos realizan solicitudes a los Servicios Web basados en SOAP.

**Vista:** incluye todas las paginas mostradas al cliente, entre las que destacan formularios, solicitudes de información, enlaces, por mencionar algunas. En este caso se utilizan archivos con terminación xhtml utilizando etiquetas del marco de trabajo JSF (Java Server Faces) así como componentes de la biblioteca de PrimeFaces.

### Arquitectura del apartado Móvil

La sección móvil se divide de la siguiente manera:

**Cliente:** es la aplicación, en este caso Aplicación Android o iOS, que realiza las peticiones por medio del protocolo HTTP (Hypertext Transfer Protocol, Protocolo de Transferencia de Hipertextos), solicitando la información que necesite.

**Vista:** está compuesta por todas aquellas páginas HTML (HiperText Markup Language, Lenguaje de Marcación de Hipertexto) con JavaScript y CSS (Cascading Style Sheets, u Hojas de Estilo en Cascada) que se despliegan hacia el cliente, como lo son solicitudes de información, direccionamiento hacia otras páginas, errores, por mencionar algunas.

**Controlador:** es el que recibe las peticiones del cliente apoyándose de JQuery y JavaScript. Se encarga de solicitar los datos al modelo y de comunicárselos a la vista.

**Modelo:** se tiene el proveedor de servicios RESTful, el cual se encarga de acceder a la información sin que el controlador se entere, con la ayuda de la biblioteca de Twitter4j recuperará y analizará los tweets de los usuarios, sin tener ningún problema, ya que para acceder a la aplicación E-learning se utiliza Twitter como medio de autenticación, obteniendo el permiso del usuario. Las tecnologías a utilizar para el desarrollo de este proyecto son las siguientes: como metodología se emplea Mobile-D, el marco de trabajo seleccionado es PhoneGap, los lenguajes de programación utilizados serán HTML5, JavaScript y CSS puesto que son los que utiliza PhoneGap por defecto para el desarrollo de aplicaciones móviles, el gestor de base de datos será PostgreSQL y por último como IDE NetBeans.

## Resultados y discusión

### Análisis de algoritmos de agrupamiento

Se realizó un análisis de los algoritmos de agrupamiento para el desarrollo del módulo de recomendación para la creación de grupos de trabajo con perfiles afines.

Los métodos de agrupamiento no paramétricos pueden dividirse en tres grupos fundamentales: jerárquicos, particionales y basados en densidad.

Los algoritmos jerárquicos son aquellos en los que se va particionando el conjunto de datos por niveles, de modo tal que en cada nivel generalmente, se unen o se dividen dos grupos del nivel anterior, según si es un algoritmo aglomerativo o divisivo [5].

**Tabla 1. Algoritmos jerárquicos.**

Algoritmo	Características
Aglomerativo	<ul style="list-style-type: none"> <li>✓ Empiezan con los elementos individuales, considerándose cada uno de ellos como un grupo.</li> <li>✓ Los objetos más similares son los primeros en unirse, y los grupos así formados continúan uniéndose en forma progresiva de acuerdo a sus similitudes.</li> <li>✓ Finalmente, todos los grupos quedan fusionados en uno sólo.</li> </ul>
Divisivo	<ul style="list-style-type: none"> <li>✓ Trabajan en el sentido opuesto al aglomerativo.</li> <li>✓ Un único grupo inicial de elementos se divide en dos subgrupos, de manera que los elementos de un subgrupo sean "bien disímiles" con los elementos del otro.</li> <li>✓ Estos subgrupos son divididos sucesivamente en subgrupos disímiles, hasta que queden tantos subgrupos como cantidad de elementos a agrupar.</li> </ul>

Los algoritmos particionales son los que realizan una división inicial de los datos en grupos y luego mueven los objetos de un grupo a otro según se optimice alguna función objetivo [6].

**Tabla 2. Algoritmos particionales.**

Algoritmo	Características
K-Means	<ul style="list-style-type: none"> <li>✓ Es uno de los más simples y conocidos algoritmos de agrupamiento.</li> <li>✓ Sigue una forma fácil y simple para dividir una base de datos dada en k grupos (fijados a priori).</li> <li>✓ La idea principal es definir k centroides (uno para cada grupo) y luego tomar cada punto de la base de datos y situarlo en la clase de su centroide más cercano.</li> <li>✓ El próximo paso es recalcular el centroide de cada grupo y volver a distribuir todos los objetos según el centroide más cercano.</li> <li>✓ El proceso se repite hasta que ya no hay cambio en los grupos de un paso al siguiente.</li> <li>✓ La desventaja que presenta es que fallan cuando los puntos de un grupo están muy cerca del centroide de otro grupo, también cuando los grupos tienen diferentes tamaños y formas.</li> </ul>
CURE	<ul style="list-style-type: none"> <li>✓ Es un algoritmo híbrido entre los dos enfoques jerárquico y particional.</li> <li>✓ Consiste en emplear las ventajas de ambos y de eliminar las limitaciones.</li> <li>✓ La similitud entre dos grupos se mide por la similitud del par de puntos representativos más cercanos, uno de cada grupo.</li> </ul>

Los algoritmos basados en densidad enfocan el problema de la división de una base de datos en grupos teniendo en cuenta la distribución de densidad de los puntos, de modo tal que los grupos que se forman tienen una alta densidad de puntos en su interior mientras que entre ellos aparecen zonas de baja densidad [6].

**Tabla 3. Algoritmos basados en densidad.**

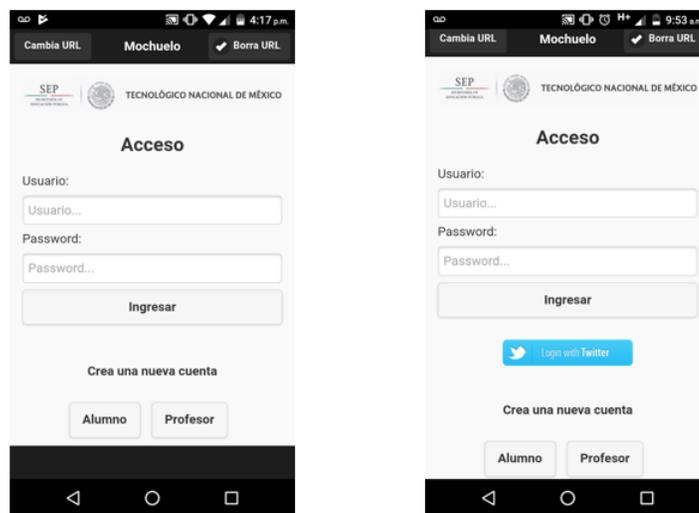
Algoritmo	Características
DBSCAN	<ul style="list-style-type: none"> <li>✓ Es el primer algoritmo basado en densidad.</li> <li>✓ Comienza seleccionando un punto p arbitrario, si p es un punto central, se comienza a construir un grupo y se ubican en su grupo todos los objetos denso-alcanzables desde p.</li> <li>✓ Si p no es un punto central se visita otro objeto del conjunto de datos.</li> <li>✓ El proceso continúa hasta que todos los objetos han sido procesados.</li> <li>✓ Los puntos que quedan fuera de los grupos formados se llaman puntos ruidos, los puntos que no son ni ruidos ni centrales se llaman puntos borde.</li> </ul>

Algoritmo	Características
OPTICS	<ul style="list-style-type: none"> <li>✓ Se basa en la necesidad de introducir parámetros de entrada en casi todos los algoritmos de agrupamiento existentes que en la mayoría de los casos son difíciles de determinar.</li> <li>✓ Se basa en el algoritmo DBSCAN creando un ordenamiento de la base de datos para representar la estructura del agrupamiento basada en densidad.</li> <li>✓ Además puede hacer una representación gráfica del agrupamiento incluso para conjuntos de datos grandes.</li> </ul>
KNNCLUST	<ul style="list-style-type: none"> <li>✓ Propone utilizar como regla basada en densidad la de los k vecinos más cercanos para tratar bases de datos de alta dimensionalidad como imágenes de satélites.</li> <li>✓ En este algoritmo se determina el número de grupos de manera automática.</li> <li>✓ Necesita la entrada de un solo parámetro: la cantidad de vecinos.</li> </ul>
SNN	<ul style="list-style-type: none"> <li>✓ La motivación para la creación de este algoritmo es la existencia de bases de datos de alta dimensionalidad tales como textos y series de tiempo, así como la existencia de grupos de diferentes formas y tamaño.</li> <li>✓ Primero encuentra los vecinos más cercanos de cada punto de la base de datos y define la similitud entre cada par de puntos en términos de cuántos vecinos más cercanos los dos puntos comparten.</li> <li>✓ Usando esta definición de similitud elimina ruido y outliers.</li> <li>✓</li> </ul>

**Desarrollo del módulo de recomendación para la creación de grupos de trabajo con perfiles afines.**

En esta sección se muestra el desarrollo de la aplicación ya realizada Mochuelo Learning la cual es una aplicación tipo E-Learning para nivel superior basada en técnicas de estadística tradicional cuyo objetivo es la recolección de datos y retroalimentación a discentes.

En la figura 2 se muestra la comparación de la interfaz principal para iniciar sesión donde el usuario se autentica o crea una cuenta como alumno o profesor y así mismo se aprecia la interfaz ya con el inicio de sesión con twitter integrado.



**Figura 2. Iniciar sesión.**

Se iniciará sesión con Twitter para extraer los tweets de cada usuario y así analizarlos aplicándoles algoritmos de agrupamiento.

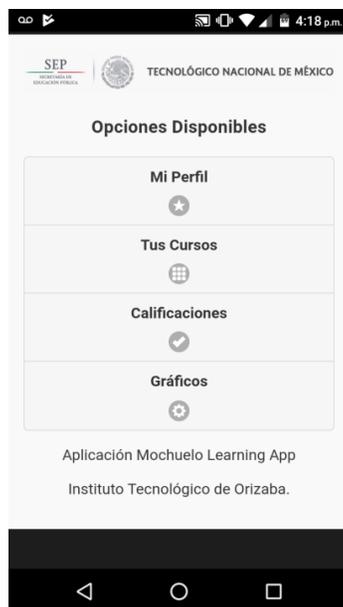


Figura 3. Pantalla Principal.

## Trabajo a futuro

Como trabajo a futuro se aplicarán los algoritmos de agrupamiento a los tweets extraídos de los usuarios que se registren en la aplicación Móvil Mochuelo Learning para la formación de los grupos de trabajo con perfiles a fines para saber en qué grupos se logra un mejor desempeño académico.

## Conclusiones

Conforme pasa el tiempo el desarrollo de aplicaciones móviles adquiere una gran importancia, ya que la mayoría de los usuarios cuenta con un teléfono inteligente que incluye redes sociales y aplicaciones tanto de entretenimiento como de aprendizaje. Los usuarios, en este caso los jóvenes, son los más involucrados con las redes sociales, como lo es Twitter, teniendo como hipótesis de partida que es una gran distracción para el aprendizaje.

En este trabajo se presentó la arquitectura de tipo E-Learning con afinidad de perfiles para crear grupos de trabajo basado en el patrón arquitectónico Modelo-Vista-Controlador para el desarrollo del Módulo Recomendación para la Creación de Grupos de Trabajo con Perfiles Afines.

## Agradecimientos

Los autores de este artículo agradecen al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo para realizar este trabajo.

## Referencias

1. "Mochuelo Learning - Apps on Google Play." [En línea]. Available: [https://play.google.com/store/apps/details?id=com.mochuelo.app&hl=en\\_US](https://play.google.com/store/apps/details?id=com.mochuelo.app&hl=en_US). [Accedido: 20/Agosto/2018].
2. "MOBILE-D" [En línea]. Disponible en: <http://agile.vtt.fi/mobiled.html>. [Accedido: 27-marzo-2018].
3. "Mobile-D" [En línea]. Disponible en: <http://www.academia.edu/23746235/Mobile-D>. [Accedido: 5-mayo-2018].
4. "Patrón de diseño Modelo-VistaControlador". [En línea]. Available: [https://www.ibm.com/support/knowledgecenter/es/SSZLC2\\_8.0.0/com.ibm.commerce.developer.doc/concepts/csdmvcdespat.html](https://www.ibm.com/support/knowledgecenter/es/SSZLC2_8.0.0/com.ibm.commerce.developer.doc/concepts/csdmvcdespat.html). [Accedido: 05/octubre/2017].
5. A. J. Soto, I. Ponzoni and G. E. Vazquez, "Análisis numérico de diferentes criterios de similitud en algoritmos de clustering", *Asociación Argentina de Mecánica Computacional*, vol. 25, 2006, pp. 993-1011.
6. D. Pascual, F. Pla and S. Sánchez, "Algoritmos de agrupamiento", *Método Informáticos Avanzados*, vol. 3, No. 1, 2007, pp. 164-174.