

# Plan de Trabajo Final - Ingeniería de Sistemas

## Perfiles de usuarios en un ambiente colaborativo de documentación de software

**Alumnos:** Rodrigo Crisanto - Martín Najle

**Directores:** Dr. J. Andrés Díaz Pace - Ing. Matías Nicoletti.

### 1. Introducción y Motivación

Durante un proyecto de software, normalmente se genera una importante cantidad de documentación (por ej.: casos de uso, arquitecturas, requerimientos, casos de test, etc.). Dicha documentación es utilizada por los distintos stakeholders del proyecto para llevar a cabo sus tareas.

El acceso de los stakeholders (tanto internos como externos a la organización) a información del proyecto es importante. Los canales de comunicación ineficientes tienden a desalentar el uso de la documentación entre los stakeholders. En el peor de los casos, la documentación podría no ser tenida en cuenta (1). Este efecto repercute negativamente en la calidad de los productos generados por las empresas de software, así como también, incrementa el esfuerzo requerido para producirlos. Por ejemplo, si la información sobre las modificaciones en el diseño de un sistema no se comunica adecuadamente al equipo de desarrolladores, como consecuencia se generaría un overhead de trabajo en el proceso de testing, o incluso se podría obtener un producto con fallas. Esta misma información de modificaciones debe también ser comunicada (probablemente con menos detalle) a los clientes del sistema.

Tradicionalmente, se han utilizado archivos de texto (por ej.: documentos MS Word) para guardar este tipo de información, pero ésta es una estrategia ineficiente cuando el volumen de documentación es significativo (2). En la actualidad, se utilizan herramientas colaborativas para tal fin. Las herramientas colaborativas están diseñadas para facilitar el trabajo en grupo, sin necesidad que todos los colaboradores se encuentren en el mismo espacio físico. El uso de herramientas colaborativas en un proyecto trae ciertas ventajas, tales como agilizar la comunicación entre los miembros y la difusión de información relevante o incentivar el trabajo en equipo y la coordinación, entre otras. Un ejemplo de estas herramientas son las Wikis (3) (4).

El uso de Wikis ha facilitado el acceso a la información, sobre todo en la Ingeniería de Software distribuida (5). No obstante, cuando el volumen de información es importante, la efectividad y utilidad del enfoque de documentación basado en Wikis conlleva desafíos. Un usuario puede experimentar dificultades para encontrar información relevante, aún cuando las Wikis ofrecen mecanismos para indexar y organizar gran cantidad de documentos.

Un enfoque utilizado habitualmente para asistir a los usuarios frente a grandes cantidades de información - situación denominada “sobrecarga de información” (6) - son las *técnicas de personalización*. Para personalizar o adaptar el comportamiento de un software, debe descubrirse el perfil de intereses de cada usuario. Un modelo de usuario almacena sus preferencias con el fin de sugerir información de utilidad y agilizar el acceso (7) (8).

Existen dos fuentes principales para recolectar los intereses de los usuarios:

- Feedback explícito: Cuando se ofrece una recomendación al usuario (por ej., un conjunto de ítems), el sistema otorga al usuario la posibilidad de calificar, dentro de un rango predefinido, los ítems de la recomendación que efectivamente ha utilizado.
- Feedback implícito: El sistema no consulta explícitamente al usuario sobre los resultados de una recomendación, sino lo infiere a partir de la interacción del usuario con el sistema sin que él lo note.

En la práctica, obtener feedback explícito es difícil ya que los usuarios no tienden a manifestar sus intereses categóricamente. Es por eso que una mejor estrategia para obtener feedback es enfocarse en las fuentes implícitas (9).

En un ambiente colaborativo, obtener feedback explícito es una tarea más sencilla de realizar, dado que se tiene el control de las acciones de los usuarios. De esta forma, la tendencia de utilizar Wikis facilita la construcción de perfiles basados en feedback implícito, con el fin de agilizar el acceso a la documentación de software.

## **2. Trabajo Propuesto**

El objetivo del presente trabajo es definir una técnica de personalización en un ambiente colaborativo de software, y brindar recomendaciones al usuario para facilitar su acceso a información específica. De esta manera, se intenta aliviar el problema de sobrecarga de información que afecta a los usuarios.

Se pretende aplicar la técnica sobre un grupo de usuarios, en un ambiente conformado por una Wiki y un chat; en el dominio de Tecnologías de la Información, características factibles de encontrar en una empresa. La interacción del usuario con el sistema será registrada a través de eventos, y se tomará en cuenta tanto la navegación del usuario sobre las páginas de la Wiki, como la comunicación con sus pares mediante el chat. Dicha interacción se monitorizará a partir de eventos, no intrusivos para el trabajo de cada usuario.

Los intereses de usuario serán obtenidos mediante técnicas de procesamiento de lenguaje natural (NLP), tomando como entrada los datos del usuario recolectados en el sistema y retornando información (por ej., en forma de tags) que describe los intereses de cada usuario. El enfoque NLP es un proceso en el cuál los datos ingresados originalmente son manipulados por distintos filtros (por ej., lematización, eliminación de stopwords) (10) (11). En este contexto, las recomendaciones se pueden realizar comparando los intereses de los usuarios, contra el contenido relevante de las páginas de la Wiki.

En lo que refiere a la evaluación del sistema de recomendación, se utilizará un grupo de usuarios simulados con perfiles predefinidos. Se contrastarán las sugerencias de información contra los intereses reales de los usuarios (conocidos de antemano por los perfiles predefinidos). Se utilizarán métricas propias del área de Recuperación de Información y Aprendizaje Supervisado, como precision, recall, accuracy y f-measure. (12).

### **3. Actividades**

- Relevamiento bibliográfico en las áreas de Construcción de perfiles de usuario, Procesamiento de lenguaje natural y Documentación de Software en Wikis. Tiempo estimado: 2 meses
- Implementación de una herramienta que combine Wiki y chat. Tiempo estimado: 3 meses
- Realización de pruebas y simulaciones. Tiempo estimado: 2 meses
- Redacción de un informe del trabajo realizado (en forma paralela con las actividades anteriores). Tiempo estimado: 6 meses.

#### 4. Bibliografía

1. *Software architecture awareness in long-term software product evolution.* **Y., Unphon H. and Dittrich.** s.l. : Journal of Systems and Software 83, 11, 2010, pp. 2211-2226.
2. *Ontology-based Software Architecture Documentation.* **de Graaf, K. and Tang, A. and Liang, P. and van Vliet, H.** s.l. : IEEE Computer Society, 2012, Proceedings Joint Working Conference on Software Architecture & 6th European Conference on Software Architecture (WICSA/ECSA), pp. 315-319.
3. *Using wikis in software development.* **Louridasm, P.** s.l. : IEEE Software, 23(2)., 2006, pp. 88-91.
4. *Using wikis to support architectural knowledge management in global software development.* **Clerc, Viktor and de Vries, Edwin and Lago, Patricia.** New York, NY, USA. : ACM, 2010, Proceedings of the 2010 ICSE workshop on Sharing and Reusing Architectural Knowledge, pp. 37-43.
5. *Experiences with a Wiki to support architectural knowledge sharing.* **Vliet, Rik Farenhorst and Hans van.** Porto, Portugal : s.n., 2008, Proceedings of the 3rd Workshop on Wikis for Software Engineering (WIKI4SE).
6. *Agents that reduce work and information overload.* **Maes, Pattie.** 1994.
7. *Recommender Systems.* **Resnick, P. Varian, H.** s.l. : Communications of the ACM 40(3), 1997, pp. 56-58.
8. *Learning user interests for user profiling in personal informations agents: Thesis.* **D., Godoy.** s.l. : IOS Press, 2006, AI Commun, pp. 391-394.
9. *Implicit feedback for inferring user preference: A bibliography.* **Kelly, D. Teevan, J.** s.l. : SIGIR Forum, 37(2), pp. 18-28.
10. **OpenNLP, Apache.** [Online] <http://opennlp.apache.org>.
11. **Group, Stanford NLP.** [Online] 2005.
12. *Evaluation: From Precision, Recall and F-Factor to ROC.* **Powers, David M. W.** s.l. : Informedness, Markedness & Correlation, 2011.
13. *Intelligent user profiling.* **Amandi, S. Schiaffino and A.** s.l. : Artificial Intelligence: an international perspective., 2009, Vol. 1, pp. 193-216.

14. *PolyLens: A recommender system for group of users.* **M. O' Connor, D. Coslay, J. Konstan, J. Riedl.** 2001, Proc. 7th European Conf. on Computer-Supported Cooperative Work, The Netherlands.

15. *Retrieval System Evaluation Using Recall and Precision Problems and Answers.* **Vijay V. Raghavan, Peter Bollman, Gwang S. Jung.** s.l. : SIGIR, 1989, pp. 59-68.

16. *Collaborations Tools.* **Cyprien Lomas, Michael Burke and Carie L. Page.** 2009.

17. *Review and functional classification of collaborative systems.* **Bafoutsou, Georgia and Gregoris Mentzas.** s.l. : International journal of information management 22.4, 2002, pp. 281-305.