



“2015, Año del Generalísimo José María Morelos y Pavón”.
 “2015, Año del Empleo”.

Moctezuma, Sonora, a 18 de marzo de 2015.
 Oficio R-058-15.

Asunto: Respuesta a Observaciones al POA 2014, Segundo Semestre.

C.P. RAFAEL PACHECO SOTO

Titular del Órgano de Control y Desarrollo Administrativo
 Presente.

En atención a Oficio OCDA/US-016/15, de fecha 10 de marzo de 2015, donde dio a conocer las observaciones derivadas de la revisión al segundo semestre del Programa Operativo Anual (POA) 2014. De la manera más atenta, me permito anexar justificación enviada por el Secretario General Académico (SGA) con Mem. No. SGA/05-15, donde se envía además, la evidencia del cumplimiento de la meta 26 “Capacitación de Profesores en su área de desempeño”, y de la meta 32 “Apoyo y desarrollo para incrementar el número de PTC con perfil PROMEP”, pertenecientes a la Secretaría General Académica donde se operan ambas metas. También, se adjunta MEM-R-002/15 donde se instruye al SGA a fin de que se tomen las medidas pertinentes para evitar recurrencias.

Lo anterior, con el fin de dar cumplimiento y solventar en su caso, las observaciones realizadas por Usted a esta Institución.

Para cualquier aclaración o ampliación de información sobre este particular, me reitero a su distinguida consideración.

ATENTAMENTE

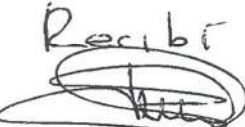
“Una Puerta Abierta al Cambio y al Desarrollo”


ING. GABRIEL AMAVIZCA HERRERA
 Rector.

C.e.p.- Mtra. Ana Judith Barceló Moreno, Jefe de la Unidad de Planeación, Presente.
 C.e.p.- Archivo

GAH/*amdv*



Recibido

 18/03/2015



SEC





“2015, Año del Generalísimo José María Morelos y Pavón”.

“2015, Año del Empleo”.

Moctezuma, Sonora, a 17 de marzo de 2015.

Mem. No. SGA/05-15.

Asunto: Atención a observaciones del POA 2014, segundo semestre.

ING. GABRIEL AMAVIZCA HERRERA

Rector.

Presente.


De la manera más atenta, me permito adjuntar evidencia documental de la meta 26 “*capacitación de Profesores en su área de desempeño*” y 32 “*apoyo y desarrollo para incrementar el número de PTC con perfil PROMEP*” pertenecientes al Programa Operativo Anual (POA) 2014 de la Secretaría General Académica que dirijo. Cabe destacar, que no se demostraron dichas evidencias al momento de la presentación del informe ante la Unidad de Planeación, debido a una falta de comunicación por el personal a mi cargo. Sin embargo, a la meta 26 se le dio cumplimiento desde el segundo trimestre ya que se acordó con los docentes que por economías al presupuesto se acudiera a capacitación a lugares más cercanos que no implicaran mayor costo, tanto para los docentes, como para la Institución, ya que originalmente se había planteado se acudiera a Los Cabos, B.C. en el segundo semestre del 2014. No obstante, se cumplió con la capacitación planteada.

En lo referente a la meta 32, se realizó una publicación en el primer trimestre y otra en el segundo trimestre, y por un error involuntario no se reportó ni se presentó la evidencia ante quienes integran el informe en la Universidad de la Sierra.

Le informo que el personal a mi cargo y un servidor, nos comprometemos a que en lo sucesivo se entablará una comunicación más estrecha con el área encargada de integrar los informes trimestrales para evitar recurrencias.

Sin otro particular, aprovecho la ocasión para enviarle un cordial saludo y reiterarme a sus órdenes.

ATENTAMENTE


LIC. JULIÁN MORENO BARCELÓ
Secretario General Académico

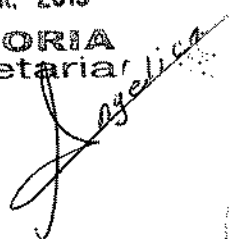
C.c.p. - Archivo

JMB/*fvnr*

UNIVERSIDAD DE LA SIERRA
RECIBIDO

18 MAR. 2015

RECTORIA
Secretaría





SEC

UNIVERSIDAD DE LA SIERRA | CARRETERA MOCTEZUMA-CUMPAS NM. 2.5 | TEL +52 (634) 342-96-00
MOCTEZUMA, SONORA, MÉXICO | www.unisierra.edu.mx





Mexican Association of
Teachers of English to
Speakers of Other Languages

2nd Regional CONVENTION

SONORA NORTE

Awards the Following

CERTIFICATE OF PARTICIPATION

to

Salomé Frisby

For attending the 2nd Regional Convention Sonora Norte-Sur
held at Tec de Monterrey Campus Sonora Norte
12 hours

M. Ed. Mayael Magdaleno Moreno
SONORA NORTE PRESIDENT

Hermosillo, Sonora Mexico
April 4th and 5th, 2014

Mtra. Clarissa Flores Chong
SONORA SUR PRESIDENT



Mexican Association of
Teachers of English to
Speakers of Other Languages

2nd Regional Convention

Awards the Following CERTIFICATE OF PARTICIPATION

to

Diana Julissa Zepeda Vázquez

For attending the 2nd Regional Convention Sonora Norte-Sur
held at Tec de Monterrey Campus Sonora Norte
12 hours


M. Ed. Mayra del Magdalena Moreno
SONORA NORTE PRESIDENT


Mtra. Clarissa Flores Chong
SONORA SUR PRESIDENT

Hermosillo, Sonora México
April 4th and 5th, 2014



Mexican Association of
Teachers of English to
Speakers of Other Languages

2nd Regional Convention

Awards the Following

CERTIFICATE OF PARTICIPATION

to

Alejandra Frisby Morales

For attending the 2nd Regional Convention Sonora Norte-Sur
held at Tec de Monterrey Campus Sonora Norte
12 hours

M. Ed. Mayael Magdaleno Moreno
SONORA NORTE PRESIDENT

Hermosillo, Sonora México
April 4th and 5th, 2014

Mtra. Clarissa Flores Chong
SONORA SUR PRESIDENT

García Gorrostieta, Jesús Miguel; González López, Samuel; López López, Aurelio
Tutor Inteligente para Propuestas de Investigación
Conciencia Tecnológica, núm. 47, enero-junio, 2014, pp. 43-48
Instituto Tecnológico de Aguascalientes
Aguascalientes, México

Disponible en: <http://www.redalyc.org/articulo.oa?id=94421297005>



Conciencia Tecnológica,
ISSN (Versión impresa): 1405-5597
contec@mail.ita.mx
Instituto Tecnológico de Aguascalientes
México

Tutor Inteligente para Propuestas de Investigación Investigación

M.C. Jesús Miguel García Gorrostieta¹, M.C. Samuel González López², Dr. Aurelio López López²

¹ Universidad de la Sierra, Carret. Moctezuma-Cumpas Km. 2.5, C.P. 84560 Moctezuma, Sonora, Tel. 01 (634) 3429600, jmgarcia@unisierra.edu.mx.

² Instituto Nacional de Astrofísica, Óptica y Electrónica, Luis Erro 1, Santa María Tonantzintla, C.P. 72840, San Andrés Cholula, Puebla, Tel. 01 (222) 2663100, {sgonzalez, allopez}@inaoep.mx

Resumen

Estudiantes de Nivel Licenciatura llevan a cabo proyectos de final de curso en los cuales se solicita la estructuración de un trabajo de investigación. Con el fin de mejorar la calidad de dicha tarea y guiar en el desarrollo de este tipo de trabajo, se implementó un Sistema Tutor para ayudar a los estudiantes. En este artículo se presenta un Tutor Inteligente basado en la web para proporcionar asesoramiento a los estudiantes en la estructuración de borradores de proyectos de investigación. Proponemos un modelo del estudiante basado en una red para seguir el progreso de cada estudiante en el desarrollo del proyecto, así como una retroalimentación personalizada en cada evaluación. El tutor incluye un módulo para evaluar la riqueza léxica, que se realiza en términos de densidad léxica, diversidad léxica y sofisticación. Se presentan los resultados de un estudio piloto con estudiantes que usaron el sistema, además de un análisis del cuestionario con el cual se realizó la evaluación empírica, en la cual se muestra que los estudiantes encontraban a la herramienta útil para mejorar sus escritos.

Palabras clave: E-learning, Procesamiento del Lenguaje Natural, Sistemas de Tutoría Inteligente, Riqueza Léxica.

Abstract

College students elaborate end-of-course project proposals which have the structure of a research paper. In order to improve the quality of the project and guide them through the development of this type of work, a Tutor System was implemented to help the students. In this paper we present a web-based Intelligent Tutor System to provide advice to students in the drafting of research project proposals. We propose a student model based on a network to track the progress of each student in the development of the project and provide a personalized feedback on each assessment. The tutor includes a module for assessing the lexical richness, which is done in terms of lexical density, lexical

variety and sophistication. We present the results of a pilot study with students which used the system, and the analysis of a questionnaire used to perform an empirical evaluation, that showed that students found that this tool was useful to improve their writings.

Key words: E-learning, Natural Language Processing, Intelligent Tutoring System, Lexical Richness.

Introducción

El realizar trabajos de investigación bien estructurados es una labor compleja que requiere del apoyo del profesor. Para dicha tarea, tecnologías tales como los sistemas de tutoría pueden ser la solución. Un sistema de tutoría inteligente (ITS) es un sistema que ofrece una instrucción y retroalimentación personalizada a los estudiantes sin la participación del profesor. Avances en los ITS incluyen el uso de las tecnologías del lenguaje natural para realizar la evaluación automática de la escritura y proporcionar retroalimentación, tal como se presenta en el artículo de Crossley *et al* [1]. Writing Pal (WPal) es un ITS que ofrece estrategias y prácticas basadas en juegos para desarrollar las habilidades del proceso de escritura. También existen agentes inteligentes virtuales capaces de responder a las preguntas del estudiante en relación a un tema académico [2]. También se tienen ITS basados en el diálogo, como el llamado Guru propuesto en [3], el cual tiene un agente tutor animado que incita a la participación del estudiante en una conversación colaborativa que hace referencia a un espacio de trabajo de hipertexto, presentando y animando imágenes significativas para la conversación. Otro ITS basado en el diálogo es AUTO Tutor el cual utiliza diálogos como su principal actividad de aprendizaje [4]. Todos estos ITS utilizan el lenguaje natural para interactuar con el alumno de manera similar al ITS presentado en el presente trabajo.

En la Universidad de la Sierra se realizan proyectos de fin de curso para los cuales se les solicita a los estudiantes estructurarlos con los elementos de un

trabajo de investigación. Con el fin de mejorar la calidad de los trabajos y guiar a los estudiantes en el desarrollo de dichos trabajos se utilizó un sistema tutor inteligente para apoyar a los estudiantes. En este artículo se presenta un Tutor Inteligente basado en la web para proporcionar asesoramiento a los estudiantes en la estructuración de borradores de proyectos de investigación. Proponemos un modelo del estudiante basado en una red para seguir el progreso de cada estudiante en el desarrollo del proyecto, así como una retroalimentación personalizada en cada evaluación. El tutor incluye un módulo para evaluar la riqueza léxica, que se realiza en términos de densidad léxica, diversidad léxica y sofisticación.

Hay ciertos métodos para evaluar el uso del vocabulario de un documento. Uno de ellos es medir la sofisticación, utilizando un listado de las 3000 palabras más usadas del español [5]. Para el español, algunos estudios utilizan la lista proporcionada por la RAE (Real Academia Española) de 1000, 5000 y 15000 palabras más frecuentes. Otros trabajos utilizan Yule's K para medir la riqueza en los textos [6], este tipo de medida se centra en la repetición de las palabras y es considerado una medida de la diversidad léxica.

El proceso de redacción de proyectos de investigación por lo general, no es una tarea fácil para los estudiantes. Por lo tanto, el sistema propuesto tiene la intención de apoyar la labor del profesor, así como la de guiar a los estudiantes a través de dicho proceso. También aplicamos una evaluación empírica con los estudiantes para verificar la eficacia del sistema propuesto y se presentan sus resultados.

Fundamentación Teórica

El tutor inteligente presenta el material referente a los diferentes elementos del proyecto, tales como el planteamiento del problema, hipótesis, objetivos y justificación en el Módulo de Dominio. Para cada elemento, se aplica un cuestionario para verificar la lectura de materiales y se realizan ejercicios prácticos usando el Analizador Léxico para lograr un nivel aceptable en la densidad, diversidad y sofisticación en los textos del estudiante. Los resultados del cuestionario y del análisis léxico se envían al módulo de Progreso del Estudiante para actualizar el estado del conocimiento del alumno en una red. La figura 1 muestra el modelo de tutor inteligente. El Módulo de Progreso del Estudiante (MPE) registra el progreso del estudiante en una red como la que se ilustra en la figura 2. Cuando el estudiante termina el cuestionario, el valor del nodo del elemento cuestionario se actualiza y el MPE calcula el progreso del estudiante para el nodo padre utilizando los pesos asignados a cada pregunta del cuestionario [7].

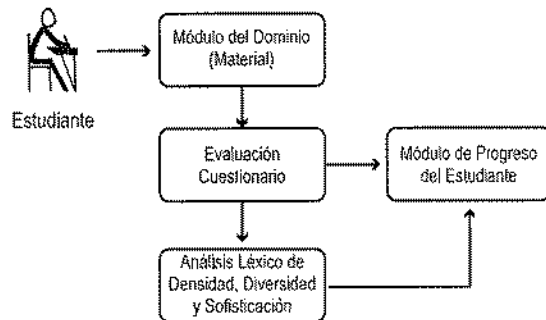


Fig. 1. Modelo del Sistema Tutor Inteligente

De manera similar, a la hora de realizar los ejercicios con el analizador léxico, el nodo correspondiente en la red se actualiza y el MPE estima el progreso del estudiante para el nodo padre utilizando los pesos asignados a la densidad léxica, la diversidad y sofisticación en el Analizador Léxico.

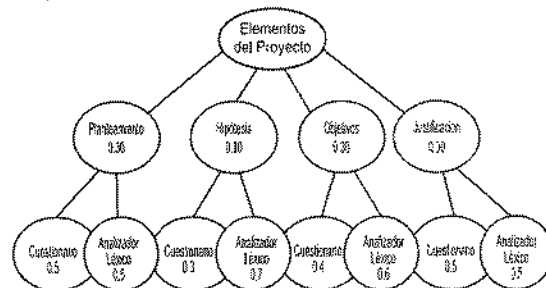


Fig. 2. Red usada en el Módulo de Progreso del Estudiante

La figura 2 presenta los pesos asignados a cada nodo a partir de la experiencia como instructores. Por ejemplo, en el nodo de cuestionario del objetivo, se asigna un peso del 40% de avance del nodo padre, en dicho cuestionario se incluyen 5 preguntas para verificar que el estudiante ha leído el material. Una vez que el estudiante ha contestado correctamente a las preguntas, esto le permite pasar a utilizar el analizador léxico para realizar tres ejercicios que tienen un peso combinado de 60% del nodo padre, que se distribuye de la siguiente manera: 20% para la densidad léxica, 20% para la diversidad léxica, y, finalmente, el 20% para la sofisticación.

En la Figura 3 se muestra el modelo de analizador léxico, el análisis léxico se centra en la evaluación de tres medidas: la densidad léxica, la diversidad léxica y sofisticación, que en conjunto evalúan la riqueza léxica. La primera medida, la densidad léxica tiene como objetivo reflejar la proporción de palabras de contenido en el texto completo. Esta medida se calcula dividiendo los tipos léxicos únicos o palabras de contenido (Tlex) por el total de palabras del texto evaluado (N), es decir, el número de palabras antes de quitar las palabras vacías [8].

La diversidad léxica busca medir la capacidad del alumno para escribir sus ideas con un vocabulario variado. Esta característica se calcula dividiendo los tipos léxicos únicos (Tlex) por el total de tipos léxicos (NLEX). Tlex se refiere los términos únicos de contenido, mientras que NLEX representa términos totales del contenido, ignorando las palabras vacías. La tercera medida es la sofisticación, que pretende indicar el conocimiento de conceptos técnicos y es la proporción de palabras “sofisticadas” empleadas. Esta medida se calcula como el porcentaje de las palabras no incluidas en la lista de 1000 palabras comunes, proporcionadas por la Real Academia Española¹. Todas las medidas toman valores entre 0 y 1, donde 1 indica un alto valor léxico, y valores cercanos a cero significa un valor bajo del léxico en la sección evaluada.

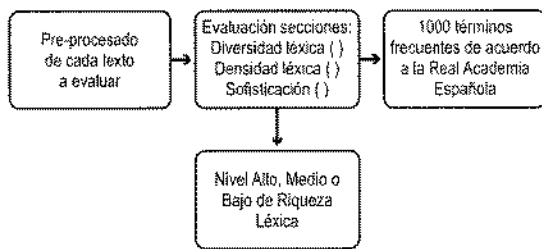


Fig. 3. Modelo del Analizador Léxico

El pre-procesamiento del texto involucra el filtrado y eliminación de palabras vacías de una lista proporcionada por el módulo de NLTK-Snowball². Las palabras vacías son las preposiciones, conjunciones, artículos y pronombres. Después de este paso, sólo palabras de contenido se mantienen, lo que permite el cálculo de las tres medidas. Por último, los resultados producidos por el Analizador Léxico se envían al módulo de progreso del estudiante, para que el tutor inteligente procese los resultados obtenidos por el usuario.

Se definió una escala para los resultados del analizador léxico con los valores de Alta, Media y Baja riqueza léxica establecidos sobre la base de nuestro trabajo previo [9] [10], donde se analizaron las propuestas de investigación y tesis a distintos niveles.

Materiales y Métodos

El sistema tutor inteligente fue desarrollado con PHP y MySQL con el paquete XAMPP para tener un acceso a la web, el analizador léxico es desarrollado en Python debido a la facilidad de acceso de dicho lenguaje a las herramientas de procesamiento del lenguaje natural. El analizador utiliza la herramienta

de código abierto FreeLing³ para el análisis de las palabras y posteriormente realizar el análisis de densidad, diversidad y sofisticación del texto. La figura 4 muestra la interfaz gráfica del sistema en el cual se observa el enlace INICIO en el menú principal para acceder a los elementos del proyecto, en el interior nos encontramos con enlaces para acceder al planteamiento del problema, hipótesis, objetivos y justificación. Para cada elemento, hay tres secciones: materiales, cuestionario y evaluación práctica. En esta figura, también podemos observar la sección AVANCE en el lado inferior izquierdo, donde se presenta los porcentajes de progresos en el concepto con el 70% y el 21% del curso completo. Como podemos observar, para entrar en la evaluación práctica, el estudiante primero debe completar con éxito un cuestionario de conocimientos básicos del concepto.

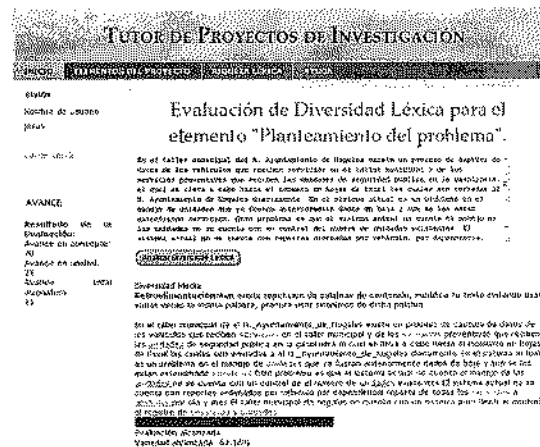


Fig. 4. Analizador Léxico para Diversidad

La sección de la evaluación práctica está presente en la figura 4, donde el alumno escribe el planteamiento del problema a analizar, primeramente el análisis de densidad mide el equilibrio entre las palabras de contenido y las palabras vacías; si el texto tiene demasiadas palabras vacías se tendrá una densidad léxica baja. A continuación, el análisis de diversidad, en el cual se mide la repetición de palabras de contenido, como se observa en la Figura 4 se repiten las palabras “servicios” y “unidades” etiquetadas de color rojo. Este caso se tiene un nivel medio de diversidad léxica y con una retroalimentación al estudiante que le plantea: “Aún existe repetición de palabras de contenido, modifica tu texto evitando usar varias veces la misma palabra, procura usar sinónimos de dicha palabra” con un 62.16 % de diversidad, que es gráficamente ilustrado por la barra de progreso en la parte inferior de la figura.

¹ <http://corpus.rae.es/frecuencias.html>
² <http://nltk.org/>

³ <http://nlp.fsi.upc.edu/freeling/>

En la figura 5 se presenta el análisis léxico para la densidad donde se observa en color rojo las palabras vacías y se indica que se tiene una densidad BAJA debido a que se tienen más palabras vacías que palabras de contenido con un 50.98% de densidad léxica, se envía al alumno la siguiente recomendación: "Se sugiere revisar el texto, ya que existen pocas palabras de contenido, procura reducir los términos en subrayado rojo".

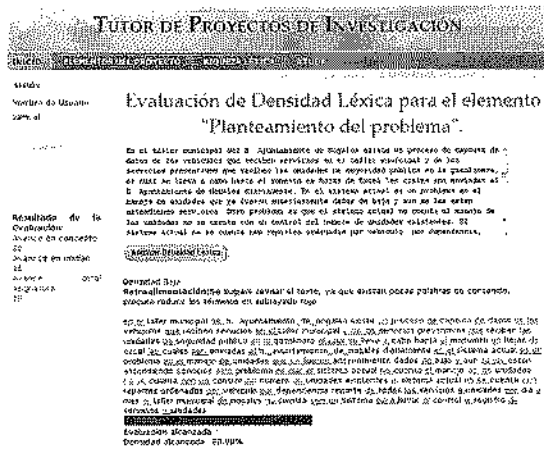


Fig. 5 Análisis léxico de densidad

Al finalizar el ejercicio de densidad léxica, el alumno puede acceder al ejercicio de la sofisticación que mide el grado en que el estudiante usa palabras poco comunes, como son el caso de las palabras técnicas del dominio de la informática.

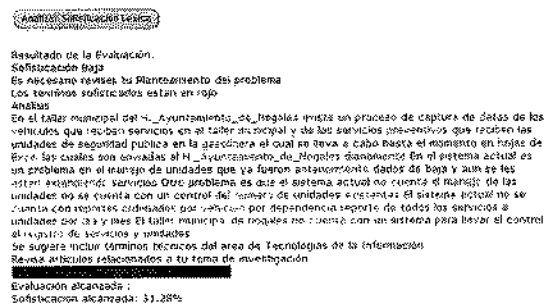


Fig. 6 Análisis Léxico de Sofisticación

En la figura 6 se observa el analizador léxico de sofisticación en el cual se marcan de color rojo las palabras que se consideran sofisticadas. En la imagen se puede observar el resultado del analizador léxico indicando una sofisticación BAJA con un 51.28%, con la recomendación de que "Se sugiere incluir términos técnicos del área de Tecnologías de Información, Revisa artículos relacionados a tu tema de investigación".

Una vez completado los tres análisis léxicos, el estudiante puede pasar al siguiente elemento del proyecto y el profesor puede revisar un planteamiento del problema más refinado.

La herramienta utilizada para verificar el nivel de aceptación de los usuarios, fue un cuestionario con los niveles de 1 = Totalmente en desacuerdo (TD), 2 = En desacuerdo (D), 3 = Neutral (N), 4 = De acuerdo (A), 5 = Muy de acuerdo (MA) basados en el modelo TAM (Technology Acceptance Model) [11].

La utilidad del Sistema

1. ¿El sistema mejora la eficacia de mi aprendizaje?
2. ¿El sistema mejora el rendimiento de mi aprendizaje?
3. ¿En general, creo que el sistema es una ventaja para mi aprendizaje?

Facilidad de Usar el Sistema

1. ¿Aprender a usar el sistema es fácil para mí?
2. ¿El proceso de usar el sistema es claro y entendible?
3. ¿En general, creo que el sistema es fácil de usar?

Adaptabilidad del Sistema

1. ¿El contenido proporcionado es adecuado para mi nivel actual de conocimiento?
2. ¿El apartado de porcentaje de avance del tema mejora tu motivación para terminar el tema y la unidad?
3. ¿Los ejemplos proporcionados en el material facilitaron la redacción de los elementos del proyecto?
4. ¿El material proporcionado en el sistema es completo y claro para permitirme realizar la redacción de los elementos del proyecto?
5. ¿El cuestionario me permitió evaluar instantáneamente mi comprensión sobre el tema y avanzar en el mismo?
6. ¿La retroalimentación del cuestionario (pistas) me ayudaron a identificar las áreas problemáticas?
7. ¿La retroalimentación del cuestionario (pistas) me ayudó a resolver las preguntas de manera eficaz?
8. ¿La retroalimentación (pistas) del analizador léxico de los elementos del proyecto me ayudó a mejorar mi redacción?
9. ¿La retroalimentación del analizador léxico de los elementos del proyecto (texto en rojo) me ayudó a identificar las palabras que se pueden mejorar?
10. ¿La retroalimentación del analizador léxico de los elementos del proyecto me ayudó a escribir dichos elementos?

La intención de utilizar el Sistema

1. ¿Tengo la intención de utilizar el sistema en el futuro como un método alternativo de aprendizaje aparte del material de clase?
2. ¿Tengo la intención de utilizar el Sistema en el futuro como un método alternativo de aprendizaje en caso de tener algún problema de comprensión del tema enseñado en clase?

Como se puede observar en la encuesta se cubren los cuatro aspectos del modelo TAM [11] con respecto a la utilidad del sistema, facilidad de usar el sistema, adaptabilidad del sistema y la intención de uso del sistema. Dicha encuesta se aplicó una vez que el alumno logró completar los apartados del planteamiento del problema, hipótesis, objetivos y justificación con los tres análisis léxicos en niveles aceptables.

Resultados y Discusión

Se aplicó una evaluación empírica para verificar la efectividad y aceptación del sistema en la Universidad de la Sierra de una Carrera afín a las Ciencias de la Computación. Se conformaron dos grupos de 14 estudiantes, a ambos grupos se les pidió el mismo producto, es decir se les solicitó redactar el anteproyecto de su trabajo final, el cual consistía en planteamiento del problema, hipótesis, justificación y objetivos. Al primer grupo denominado grupo de control únicamente se les entregó material impreso referente a la elaboración del anteproyecto y se les incentivó a consultar con el profesor las dudas referentes al trabajo. Por otro lado, el segundo grupo denominado grupo Experimental se les solicitó utilizar el sistema tutor inteligente y se les explicó la forma de usarlo. La duración del experimento fue de una semana.

Al terminar el experimento se analizaron los resultados, observándose que el grupo de control no consultó al profesor para revisar su trabajo antes de la entrega final. Al no estar obligados a realizar una revisión previa los alumnos la omitieron y solamente entregaron el trabajo al llegar la fecha de entrega. Por otro parte el grupo Experimental al utilizar el tutor inteligente consultaron más el material y al maestro para mejorar su redacción y obtener un mayor puntaje en el sistema para concluir con los todos los elementos del proyecto. Como se puede observar en la figura 7 en la cual se presenta los valores promedio del análisis léxico para densidad, diversidad y sofisticación para el planteamiento del problema del grupo experimental y el grupo de control, se puede ver que el grupo experimental obtuvo mayor puntaje.

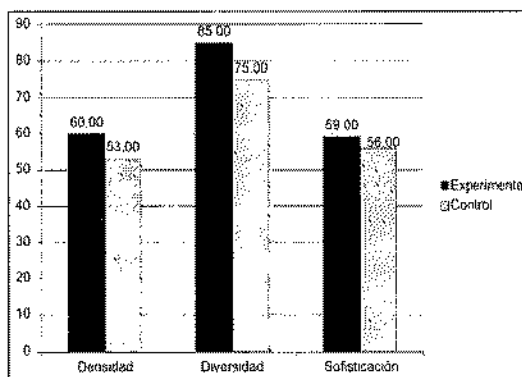


Fig. 7 Análisis Léxico - Planteamiento del Problema

Los rangos establecidos para ofrecer una retroalimentación textual en el planteamiento del problema son para un valor "bajo" menor o igual a 52%, para un valor "medio" mayor de 52% y menor de 59%, para un valor "alto" mayor o igual a 59%. De acuerdo a estos valores se observa que el grupo de control que obtuvo en promedio 53% corresponde a un valor "medio" y el grupo experimental con un 59% el cual corresponde a un valor "alto" de la densidad léxica del planteamiento del problema.

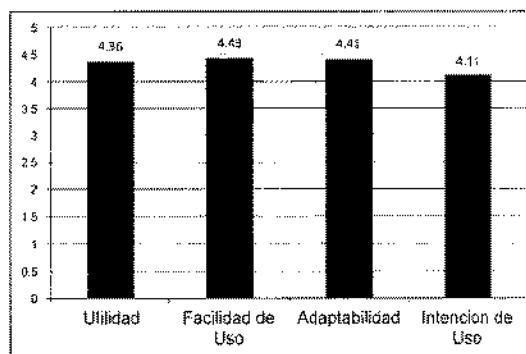


Fig. 8 Resultados de Encuesta de Satisfacción

Además, para la encuesta de satisfacción basada en el modelo TAM, los alumnos la contestaron en base a una escala de 5 puntos desde 1 como "Muy en desacuerdo" hasta '5' como "muy de acuerdo". Se observa en la figura 8 los resultados promedios por aspecto de la encuesta de satisfacción, en la cual se observa que la impresión del alumno es "de acuerdo" para todos los aspectos, con lo cual se infiere que el sistema le parece de utilidad, fácil de usar, adaptado a su nivel y tiene la intención de seguirlo usando.

En la figura 9 se pueden observar los promedios de los resultados de la encuesta de satisfacción por alumno, en la cual se aprecia que solo 2 alumnos consideraron

como la opción 3 en promedio, la cual es una respuesta neutra, el resto estuvo "de acuerdo" en considerar al sistema de utilidad, fácil de usar, adaptable y con la intención de seguirlo usando.

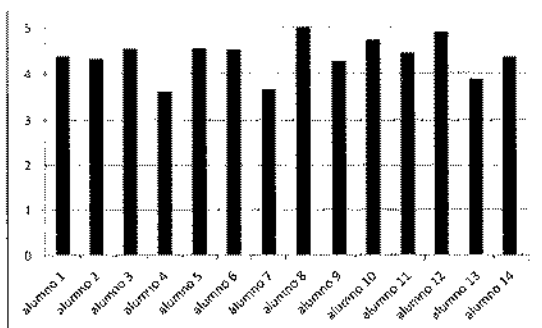


Fig. 9 Encuesta de satisfacción alumnos

Conclusiones

El uso del sistema tutor inteligente en el desarrollo de proyecto de investigación de estudiantes tiene como objetivo apoyar a los maestros en la revisión y seguimiento de los mismos, proporcionando el material necesario para el alumno en todo momento, registrando el avance del proyecto y analizando diversos aspectos del léxico de sus aportaciones, lo cual puede ser particularmente útil en las primeras etapas de la elaboración del proyecto.

Como podemos, ver el uso del ITS mejoró los tres aspectos léxico: la densidad, la diversidad y sofisticación, en el grupo experimental y de acuerdo a la encuesta de satisfacción se tiene una buena aceptación de la herramienta entre los estudiantes.

Se planea seguir extendiendo el sistema tutor inteligente con análisis adicionales, tales como revisar coherencia o aspectos particulares de las secciones centrales de la propuesta de investigación, como son los objetivos o las conclusiones.

En el trabajo futuro, tenemos la intención de poner en práctica el ITS con el uso de objetos de aprendizaje SCORM y apoyados por un LMS de código abierto para mejorar la portabilidad de los recursos digitales y mejorar la asimilación del contenido en los estudiantes. Con lo cual que se espera mejorar la estructuración de los proyectos de investigación de los estudiantes.

Referencias

[1]. Crossley, S. A., Varne, L. K., Roscoe, R. D. & McNamara, D. S. (2013), Using Automated Indices of Cohesion to Evaluate an Intelligent Tutoring System and an Automated Writing Evaluation System. In *Procs. 16th International Conference, AIED 2013*, Memphis, TN, USA. Springer, pp 269-278.

[2]. Rospide, C.G. & Puente, C. (2012), Virtual Agent Oriented to e-learning Processes, In *Procs. 2012 International Conference on Artificial Intelligence*. Las Vegas, Nevada.

[3]. Oiney, A., D'Mello, S. K., Person, N. K., Cade, W. L., Hays, P., Williams, C., Lehman, B. & Graesser, A. C. (2012), Guru: A Computer Tutor That Models Expert Human Tutors., in Stefano A. Cerrri; William J. Clancey; Giorgos Papadourakis & Kitty Panourgia, ed., 'ITS', Springer, pp. 256-261.

[4]. Graesser, A.C., D'Mello, S.K., Craig, S.D., Witherspoon, A., Sullins, J., McDaniel, B. & Gholson, B. (2008). The Relationship between Affective States and Dialog Patterns during Interactions with Autotutor, *J. Interactive Learning Research*, vol. 19, no. 2, pp. 293-312.

[5]. Schwarm, S. & Ostendorf, M. (2005), Reading level assessment using support vector machines and statistical language models. In *Procs. of the 43rd Annual Meeting on Association for Computational Linguistics (ACL '05)*, pp. 523-530.

[6]. Miranda, A. and Calle, J. (2005), Yule's Characteristic K Revisited. *Language Resources and Evaluation*, 39, 4, pp. 287-294.

[7]. Sucar, L.E. & Noguez, J. (2008), Student Modeling, in O. Pourret, P. Naim, B. Marcot (Eds.), *Bayesian belief networks: a practical guide to applications*, Wiley, pp.173-186.

[8]. Roberto, J., Martí, M. & Salamó, M. (2012), Análisis de la riqueza léxica en el contexto de la clasificación de atributos demográficos latentes. *Procesamiento de Lenguaje Natural*, No. 48, pp. 97-104.

[9]. González López, S. & López-López, A. (2012), Supporting the review of student proposal drafts in information technologies. In *Procs. of the 13th annual conference on Information technology education (SIGITE '12)*. ACM, New York, NY, USA, 215-220.

[10]. González López S. & López López A. (2014), Mining Domain Knowledge for Coherence Assessment of Students Proposal Drafts, In *Educational Data Mining: Applications and Trends*, A. Peña-Ayala (ed.), *Studies in Computational Intelligence*, Vol. 524, Springer 2014, pp. 229-255.

[11]. Tobing, V., Hamzah, M., Sura S. & Amin H. (2008), Assessing the Acceptability of Adaptive E-Learning System. In *Procs. Fifth International Conference on eLearning for Knowledge-Based Society*. December 11-12, 2008, Bangkok, Thailand, 10p.

Recibido: 12 de diciembre de 2013

Aceptado: 14 de marzo de 2014

Propuesta Metodológica para Desarrollo de Aplicaciones Móviles para Dispositivos Android

Ulises Ponce Mendoza MCS¹, ITS. Victor Yáñez Moreno², MCIE. Rafael Armando Soto Bernal³

Resumen— El desarrollo de aplicaciones para dispositivos móviles está en un fenómeno de contagio entre las empresas de desarrollo de software de tamaño PYME, mismas que se avocan a proyectos de ésta índole con metodologías probadas en el desarrollo de software para escritorio o sin ellas. Situación que pone en duda la calidad, soporte y estabilidad de las aplicaciones desarrolladas, al no existir un lineamiento general de desarrollo. Es por ello que se diseñó y probó una metodología que permitiera asegurar la calidad del software. El resultado fue una metodología derivada de un proceso incremental con implementación de características de Scrum, Kanban, Desarrollo por Prototipos y Modelo Vista-Controlador. Así mismo, se definió la infraestructura ideal de artefactos para su implementación. Su operación se probó mediante el desarrollo de una aplicación móvil en la empresa eDesarrollos de Hermosillo, Sonora.

Palabras claves—Android, Desarrollo, Metodología, Aplicaciones, Móviles.

Introducción

La mayoría de los proyectos de software que se desarrollan en PYMES tienen un elevado porcentaje de fracaso, éste puede ser completo, es decir que no se lleve a cabo la entrega del producto o parcial, es decir, que no se cumpla con los requerimientos solicitados por el cliente, o que se tengan problemas fuertes de errores en la implementación, dificultad para el mantenimiento y desfase en atención a las necesidades. Todo lo anterior redundando en costos no estimados por las empresas que impactan fuertemente en su solvencia económica y permanencia en el mercado.

Las tres variables más relevantes en el fracaso de los proyectos de software son a) Errores de estimación de Costo; b) Falla en la medición del riesgo del proyecto, y c) Desarrollo basado en programadores estrella. Especialmente en los proyectos donde la tecnología es novedosa se tienden a fallar porque no es posible medir con precisión los costos involucrados de desarrollo a largo plazo, así mismo, no tienen formas para definir qué tan viable técnicamente es el desarrollar la aplicación como el cliente la desea en el plazo acordado y finalmente muchas PYMES dependen de un desarrollador estrella que de rotación de personal compromete en gran medida las operaciones de la misma. Como mecanismo para paliar lo anterior, el uso de metodologías de desarrollo de software ha demostrado ser una excelente práctica que contribuye al cumplimiento de fechas de entrega, al mismo tiempo que asegura la calidad del software y atiende las necesidades del cliente. En especial las metodologías ágiles permiten a las PYMES de desarrollo de software responder de manera eficiente a los cambios en los requerimientos del producto en desarrollo, a la vez que mantienen el control de riesgo, sin caer en exceso de mecanismos de control. Asimismo éstas permiten una mayor flexibilidad que las metodologías tradicionales de desarrollo, que se bloquean muy pronto en los detalles del proyecto y son menos capaces de ajustarse a las cambiantes necesidades de los clientes, del mercado y de los desafíos imprevistos que plantea la tecnología.

Por otra parte el desarrollo de aplicaciones móviles está teniendo una etapa de contagio en nuestro país, sin embargo, quienes están aprovechando la creciente demanda son las empresas de desarrollo grandes y consolidadas, lo que ha redundado en un conjunto de aplicaciones de alta calidad desarrolladas por estas empresas y otro conjunto numerosamente mayor de aplicaciones desarrolladas por PYMES o freelance que tienen problemas de calidad en la operación, usabilidad y soporte, por lo que se propone el siguiente trabajo como estrategia para contribuir a la madurez del mercado de aplicaciones Android, contribuir a la mejora de la calidad de las aplicaciones desarrolladas y favorecer el desarrollo de aplicaciones en las PYMES, para hacer más competitivo el mercado.

Descripción del Método

Todo proceso de desarrollo de software requiere una metodología que permita organizar a los participantes de forma que se coordinen adecuadamente para el logro exitoso del software, es por ello, que para favorecer el éxito en el desarrollo de aplicaciones móviles se realizó un análisis basado en listas de verificación. El análisis emparejó las características de trabajo de la empresa modelo con las de varias metodologías que permitiera hacer una transición sencilla y ágil para el equipo de desarrollo, al mismo tiempo que se empataba con prácticas recomendadas de desarrollo. Para ello se realizaron observaciones in-situ durante seis semanas, en las cuales se participó como observadores y ayudantes en variados equipos de desarrollo de la empresa, y en función de las experiencias

¹ Ulises Ponce Mendoza MCS es Profesor en la División de Ingeniería y Tecnologías de la Universidad de la Sierra. upmendoza@gmail.com (autor correspondiente)

² Victor Yáñez Moreno es Ingeniero en Telemática y Sistemas, Egresado de la Universidad de la Sierra y desarrollador en la empresa eDesarrollos S.A. de C.V. vyanez@edesarrollos.com

³ Rafael Armando Soto Bernal es propietario de la empresa eDesarrollos S.A. de C.V. rsotobernal@edesarrollos.com

obtenidas y entrevistas a los desarrolladores se realizó el emparejamiento de las características con las metodologías revisadas. Dentro de los primeros hallazgos se identificaron tres roles distintos de participación en el desarrollo.

Una vez que se caracterizó el esquema de trabajo de la empresa modelo se determinaron las características de las metodologías ágiles, que eran compatibles con la empresa y que aportaran elementos de repetibilidad y control del proceso. Una vez hecho esto, se diseñó un esquema de siete etapas para el desarrollo de aplicaciones móviles.

Definida la estructura general de la metodología mediante las observaciones en el sitio y el apoyo de la literatura respectiva se seleccionó uno de los proyectos de los clientes de la empresa modelo con la finalidad de iniciar el desarrollo de la aplicación móvil utilizando la metodología propuesta. Lo anterior nos permitiría identificar con mayor precisión las buenas prácticas de cada una de las etapas de la metodología.

A pesar de ser la cuarta etapa por requerimientos propios de capacitación se decidió iniciar por determinar las mejores prácticas de codificación y la herramienta que nos ayudaría a escribir ese código, es decir el Entorno de Desarrollo Integrado (IDE). Los IDE existentes para desarrollo de aplicaciones en dispositivos móviles, en este caso con sistema operativo Android, fueron los siguientes: a) Eclipse SE con Android SDK Plug-in; b) Eclipse Android Developer Tools; c) IntelliJ IDEA con Android Plugin; y d) Google Android Studio con IntelliJ IDEA. De ellos se pudo configurar exitosamente los tres últimos y es a estos a los que se les aplicó la siguiente rúbrica de evaluación (Cuadro 1). Una vez que se emparejaron las características de cada IDE con las categorías a las que corresponde cada uno de sus componentes, se realizó un gráfico de radar para analizar gráficamente cuáles de ellos serían los más adecuados (Figura 1).

Componentes	Nivel Inadecuado	Nivel Satisfactorio	Nivel Excelente
Integración SCV	No contar con sistema de control de versiones.	Contar con plug-in Github	Contar con SCV Integrado y soporte de Git, SVN, y Mercurial.
Análisis de código	Que no importe librerías -refactorize.	Que haga refactorización y proporcione soluciones rápidas.	Proporciona una pre-visualización y Generación de recursos. Detección de errores.
Desarrollo de interfaz gráfica	No muestre una vista previa de la interfaz que está en diseño.	Muestre la interfaz y además permita hacer un "drag&drop" de componentes. Muestre reglas de alineación. Permita ver la pantalla en modo normal y modo portátil.	Que además muestre una vista previa de la pantalla en todas las resoluciones disponibles. Que permita crear dinámicamente pantallas para dispositivos con mayor resolución.
Versión	Versión Beta	Preview	Estable
Ayuda de autocompletado	No muestra opción de auto-completado al momento de escribir código.	Que muestre las opciones para autocompletar la frase.	Que además muestre una breve descripción de la variable/objeto.

Cuadro 1. Rúbrica analítica de los IDE.

Sin embargo, no sólo la herramienta de codificación es indispensable sino que el uso de un modelo para asegurar la calidad y mantenibilidad del producto final, también lo es. Por lo tanto y posterior a una investigación documental y al requerimiento implícito de desarrollo por parte de la empresa modelo de permitir que los software desarrollados tuvieran la facilidad para variar rápidamente las GUI, sin afectar los componentes de inteligencia del negocio se determinó que el Modelo Vista-Controlador (MVC) se ajustaba a dichas necesidades. Además esto le confiere más robustez a la aplicación desarrollada porque puede cumplir con el requerimiento de ser compatible con al menos las versiones Gingerbread y Jellybean que son las más usadas por los dispositivos móviles en el mercado mexicano⁴. El segundo requerimiento es el hecho de que al separar los componentes de la aplicación como lo hace MVC obtendremos ventajas al momento de dar mantenimiento y actualización a la aplicación, lo que podría redundar en más eficiencia, calidad y reducción de costos. Es importante mencionar que la empresa modelo no contaba con ningún patrón de desarrollo para móviles.

Finalmente el MVC permite seccionar de forma más fácil nuestro equipo de trabajo y dedicar una parte a desarrollo de componentes propios de la empresa, de tal forma que queden construidos módulos o librerías con funcionalidades específicas que deberían reutilizarse en proyectos posteriores, para proyectar estabilidad y madurez en los procesos de desarrollo.

Para la etapa uno se mantuvo el proceso de análisis de requerimientos con las técnicas de entrevista con el cliente y el establecimiento de acuerdos para realizar un desarrollo gradual y basado en prototipos.

Mientras que para la etapa dos al no tener ningún instrumento definido en la empresa modelo se realiza una comparativa de herramientas de prototipado rápido para el diseño de GUI, así mismo, para el caso de usabilidad y

⁴ Wayerless Tecnología Móvil. "Jelly Bean supera a Gingerbread como la versión mas usada dentro de Android". <http://www.wayerless.com/2013/07/jelly-bean-supera-a-gingerbread-como-la-version-mas-usada-dentro-de-android/> (24/02/2014)

desarrollo de interfaces de usuario se plantearon dos vertientes, la primera radica en el diseño de interfaces basado en estándares comunes como la teoría de colores y la segunda radica en utilizar los estándares que Google como empresa fundadora del sistema nos recomienda. Para éste estudio se desarrollaron dos interfaces para experimentación una con cada estándar y se sometió a prueba por parte de los usuarios de la aplicación desarrollada, se determinó un periodo de treinta días de uso para posteriormente aplicar un instrumento de validación de la calidad percibida en la interfaz. Dicho instrumento fue una encuesta que se asoció con una rúbrica de validación que incluyó las reglas de oro propuestas por Pressman.

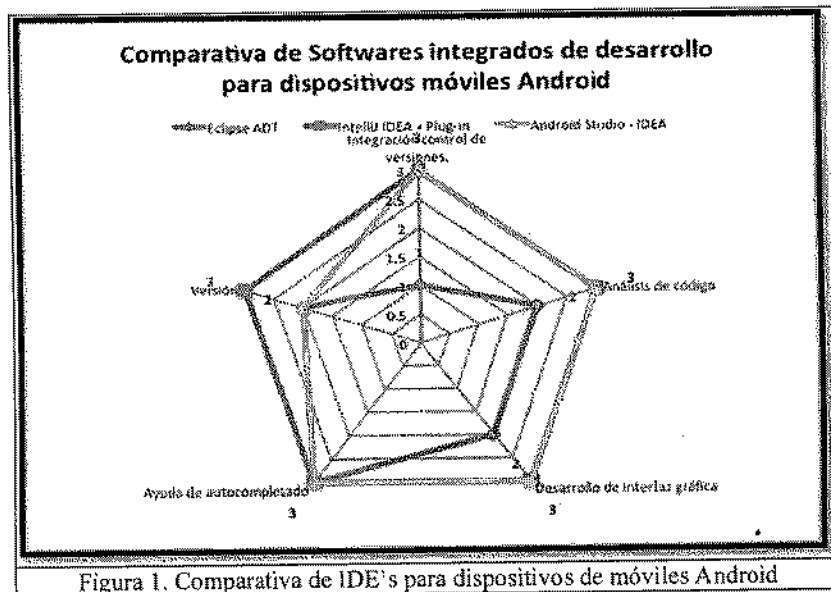


Figura 1. Comparativa de IDE's para dispositivos de móviles Android

La tercera etapa que es la definición de tareas se desarrollaba mediante un tablero Scrum en la empresa modelo, sin embargo, para agilizar el proceso y en función de la menor cantidad de desarrolladores en el equipo se propuso el uso de tableros Kanban. Finalmente la quinta y sexta etapa pertenecen a la categoría de pruebas, la primera de ellas es un conjunto determinado de pruebas de caja negra utilizando primeramente la herramienta "Monkey" incluida en el SDK de Android mediante el uso del emulador, así como, la posterior prueba en equipos reales; la segunda etapa es una prueba de percepción de funcionalidad del cliente. Una vez pasadas las pruebas y corregidos los errores, en su caso, se aprueba la última etapa de la metodología, el lanzamiento.

Resultados

Validación

Para asegurar el sustento de lo propuesto anteriormente en la metodología se llevaron a cabo los análisis de las rúbricas y otros instrumentos desarrollados obteniendo los siguientes resultados.

a) **Roles.** - Mediante observación se encontró que los roles que mejor funcionaron en los diferentes equipos de desarrollo fueron tres: Cliente, Asesor de Desarrollo, y Equipo de desarrollo. El tamaño mínimo de equipo de trabajo en la empresa debería ser de tres personas.

Cascada	Scrum	Prototipos
El equipo de desarrollo es mayor de 5 integrantes.	El equipo de desarrollo es mayor a 5 integrantes pero menor a 9.	El equipo de desarrollo puede ser menor de 3 integrantes.
Requerido contar con requerimientos funcionales antes del desarrollo.	Individuos e interacciones más que procesos y herramientas.	El cliente no identifica los requisitos detallados.
Para que el proyecto tenga éxito deben desarrollarse todas las fases.	Software que funciona más que documentación exhaustiva.	Se reducen tiempos y costos.
Las fases continúan hasta que los objetivos se han cumplido.	Responder ante el cambio más que el seguimiento de un plan.	El cliente prueba el prototipo y redefine requisitos.
La calidad del producto resultante es alta.	Colaboración con el cliente más que negociación de contratos	Existe mayor comunicación entre los desarrolladores y el cliente

Cuadro 2.- Rúbrica comparativa de metodologías contrastadas

b) **Metodología y Artefactos.** - Inicialmente se propuso que el desarrollo de software para móviles dentro de la empresa modelo se llevara a cabo siguiendo Scrum como metodología para el desarrollo de software, sin embargo, al validar los requisitos, determinamos que no era viable, siendo prototipos la que le sustituye. En la metodología por prototipos identificamos la totalidad de las características clave que fundamentan su elección. A continuación tenemos los resultados de una rúbrica elaborada en la cual se muestran las características de tres metodologías estudiadas para ser propuestas como metodología de desarrollo de software para móviles dentro de la empresa (Cuadro 2). Para determinar cuál era la mejor herramienta IDE que existe se aplicó la rúbrica diseñada. Como

podemos observar se evaluaron los componentes que se consideraron más críticos para el desarrollo de aplicaciones móviles y que ayudaran a que el programador se le facilitara el trabajo. En la Figura 1 podemos observar como

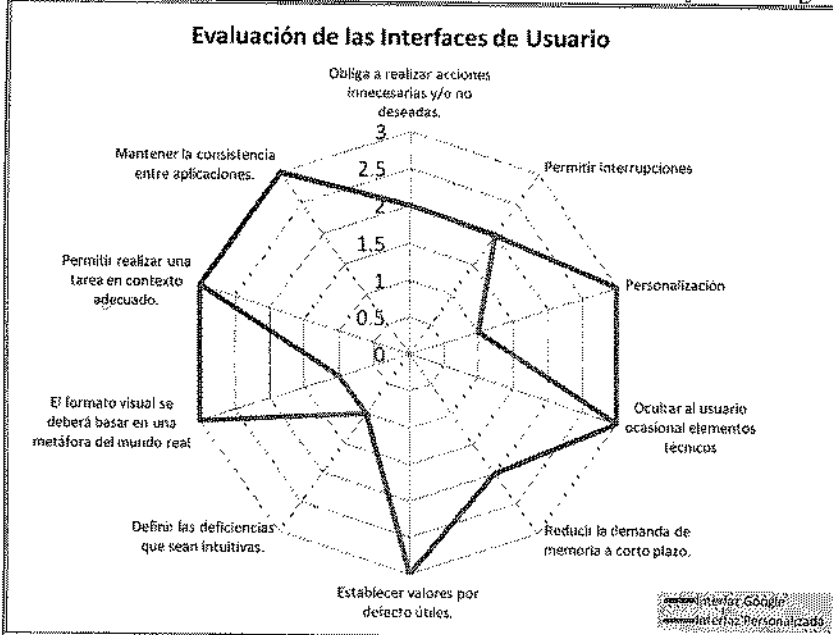


Figura 2. Evaluación de los GUI, aplicando las reglas propuestas por Pressman.

IntelliJ Idea + plugin y Android Studio cumplen con las características deseadas, sin embargo se optó por proponer el primero, porque además de ser versión estable, demostró tolerancia a fallos. Por otra parte y en base a análisis previos se determinó que en la empresa se utilizaría MVC como patrón de diseño de software móvil para asegurar la adaptación a distintas versiones del sistema operativo con un mismo núcleo de inteligencia de negocio.

Es por eso que se sugiere que para el desarrollo de aplicaciones móviles se analicen muy bien las estructuras a crear, en este caso lo modelos, los controladores y las vistas, por lo cual se recomienda realizar diagramas de clase, ya que en ellos

podemos observar de manera clara y concisa como está estructurada la aplicación. A continuación mostramos cómo se utilizaron estos diagramas para el desarrollo de una aplicación que quedaria como resultado de esta metodología.

Seguidamente en la propuesta de esta metodología llegamos al punto donde se evalúan las interfaces de usuario. Como anteriormente se estuvo mencionando, se utilizaron las tres reglas de oro que Pressman sugiere para el desarrollo de interfaces (Figura 2). En los resultados anteriores podemos observar claramente como las interfaces hacen la diferencia ante la percepción de la calidad y usabilidad del usuario, es por eso que se propone que se personalicen las interfaces que Google nos proporciona como base para el desarrollo de aplicaciones para contribuir a la usabilidad, sin descuidar la robustez. Por último tenemos los resultados de una encuesta aplicada para la evaluación de la percepción de la calidad y usabilidad ante el usuario final, teniendo los siguientes resultados. Como podemos observar algunos resultados no son los esperados, sin embargo se corroboró que en el caso de las interfaces de usuario siguiendo los estándares de Google dando un toque personalizado son la mejor opción y el modelo que se propone a seguir en la empresa eDesarrollos para el desarrollo de software para móviles en este caso con sistema operativo Android (Cuadro 3).

Preguntas	Usuario 1 Google	Usuario 2 Google + Custom	Usuario 3 Google + Custom	Persona 4 Google
¿Eran claros los iconos que se mostraban en la interfaz?	SI	SI	SI	SI
¿Eran fáciles de recordar y de invocar las acciones?	SI	NO	SI	NO
¿Cuántas funcionalidades diferentes ha utilizado?	0	2	3	2
¿Resultó fácil de aprender las operaciones básicas de la aplicación?	SI	NO	SI	SI
En comparación con otras interfaces de aplicaciones que haya utilizado, ¿Cómo evaluaría ésta?	- 50%	+10%	+10%	+1%

Cuadro 3.- Resultados de la encuesta de satisfacción del usuario

Metodología de Desarrollo Ágil de Aplicaciones Android (D3A)

De manera resumida podemos decir que la metodología consiste en: a) Roles. - El cliente, quien es el que define de una manera general los requerimientos del sistema o de la aplicación en este caso. Segundo, el asesor del proyecto, quien se encarga de plantear prototipos funcionales que a su vez son del sistema o aplicación final en base

a los requerimientos del cliente y tercero, el equipo de desarrollo, quien se encarga de realizar el prototipo funcional en base a requerimientos bien definidos. b) Etapa 1.- Esta planeación se lleva a cabo primeramente con un análisis de requerimientos general del sistema, donde el cliente y el asesor definen actividades generales, la agenda y prioridades del proyecto.

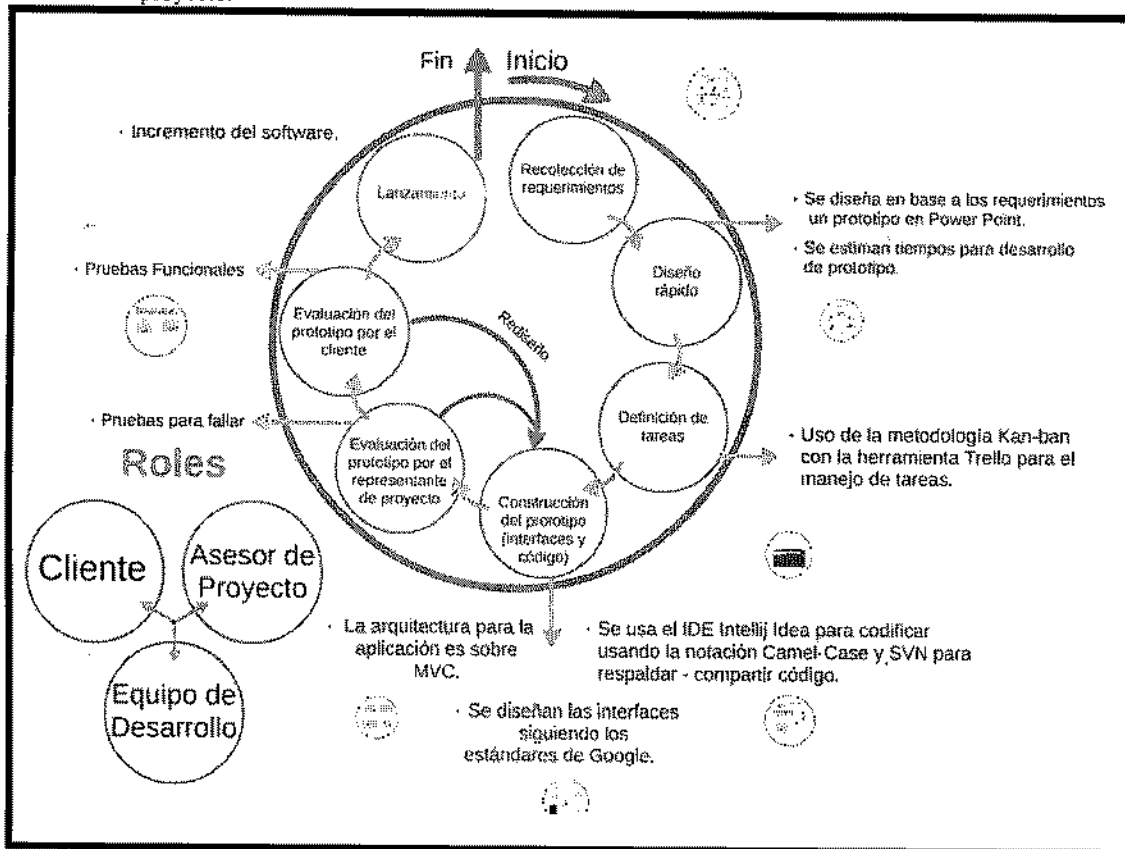


Figura 3. Metodología D3A, Desarrollo Ágil de Aplicaciones Android

c) Etapa 2.- El asesor del proyecto en base a prioridades define un primer prototipo, con aprobación del Cliente, para que el equipo de desarrollo se dé a la tarea de realizarlo. Es aquí donde se hace uso de Kanban, donde "Kan" significa visible y "ban" tarjeta, manejando un tablero muy parecido al de esta metodología, con sus ajustes. El tablero que maneja esta metodología representa de forma visual tarjetas donde se pueden observar clara y definitivamente cuales son las actividades a realizar del proyecto, y además, se observa el flujo del proyecto. El flujo del proyecto se basa en el principio del desarrollo incremental, por lo que las actividades del proyecto son divididas en distintas partes para agilizar el proceso. Normalmente cada una de esas partes se escribe en una tarjeta y se pega en el tablero, en la fase que corresponda. Dichas tarjetas deberán contener la información básica para que el equipo sepa rápidamente la carga total de trabajo que supone: normalmente descripción de la tarea con la estimación de horas. Además, se pueden emplear fotos para asignar responsables así como también usar tarjetas con distintas formas para poner observaciones o indicar bloqueos. Al final, el objetivo de la visualización es clarificar al máximo el trabajo a realizar, las tareas asignadas a cada equipo de trabajo, así como también las prioridades y la meta asignada. Es aquí donde recomienda el uso del tablero "Trello" con la siguiente configuración: Primeramente del lado izquierdo tenemos la columna "Pendientes" que es la actividad que hay que desarrollar, donde se deberán asignar etiquetas de colores que representan la prioridad y además donde si existieran recursos de apoyo para el entendimiento de la tarea a realizar aquí se incluirían. Seguidamente tenemos la columna "Development" donde esta fase es la que está en producción por parte del equipo de desarrollo. Si existieran dudas, se puede hacer una notificación al asesor del proyecto para que este trate de aclarar la situación presentada. Continuando con la visualización del tablero nos encontramos con la columna "Testing" donde se encuentran las fases, partes o prototipos que ya han sido desarrollados por parte del equipo de programadores de la empresa. Es en esta fase donde el asesor analiza el prototipo desarrollado para ver si cumple con los requerimientos para posteriormente presentarlo como

avance al cliente. Si existiera algún error, se regresa al equipo de programadores para corregir. Por último tenemos la columna "Done" donde se encuentra ya el prototipo que pasó las pruebas y está listo. Es aquí donde se decide si hay que continuar con la siguiente fase o prototipo.

Para favorecer el trabajo en equipo, se recomienda la orientación del desarrollo a tareas de resolución a corto plazo. Sin duda, otro punto clave que se consideró para establecer una metodología de desarrollo de aplicaciones es considerar la parte de escritura del código, nombramiento de paquetes, clases, variables, etc. Es por eso que se estableció que hay que documentar aunque sea muy básico como es que funciona cierta parte del código y escribir este en el estándar "CamelCase".

El desarrollar una interfaz clara y entendible debe de ser uno de los principios a seguir en el desarrollo de software para móviles, es por eso que se propone dentro de esta metodología que se sigan los estándares de Google para el diseño de interfaces, dando un ligero toque personalizado a dichas interfaces para que le sean de más agrado al usuario, y que le ayuden a entender claramente que es lo siguiente o que debería hacer en esa interfaz.

Es aquí donde ya tenemos un prototipo funcional listo para presentarlo al cliente, dando así la oportunidad de decidir el avance hacia el siguiente prototipo, ya que es un desarrollo incremental.

Véase el diagrama integrado de la metodología al final del documento.

Comentarios Finales

Conclusiones

Este trabajo pretende sentar las bases para la definición de estándares mínimos de calidad en el desarrollo de aplicaciones móviles, especialmente en aquellas PYMES que desean adentrarse en la programación de móviles, sin llegar a crear una obligatoriedad de características de las aplicaciones para no mermar la diversidad de opciones en el mercado libre de aplicaciones, pero sí con la finalidad de disminuir la cantidad de aplicaciones que incorporan fallos y/o deficiencias en la usabilidad de sus interfaces gráficas de usuario.

Asimismo, estamos convencidos que el futuro del desarrollo de software lo constituyen las PYMES por la flexibilidad con la que pueden operar al crear alianzas temporales o incorporar desarrolladores libres, por lo que la metodología propuesta incluye herramientas de uso libre para no limitar su aplicación. Además lo anterior creemos que les aportará beneficios operativos y de costos para contribuir a ampliar los márgenes económicos de operación de las PYMES de software.

Recomendaciones

Para continuar con el avance de ésta área del conocimiento sería muy recomendable iniciar con la validación de la metodología en diferentes empresas medianas y pequeñas que deseen realizar proyectos de desarrollo móvil para plataforma Android, para ir acumulando una base de datos que nos permita definir un indicador de éxito de la metodología o de la necesidad de adecuaciones. Posteriormente y en caso de ser exitosa se requiere que se realicen pruebas de escalabilidad, es decir, que tan operativo resulta en equipos más grandes de desarrollo.

Referencias

- Aguilar, Luis Joyanes. "Programación Orientada a Objetos". 2da Edición. España. Ed. McGraw-Hill, 1998.
- Bahit, Eugenia. "POO y MVC en PHP, El paradigma de la Programación Orientada a Objetos en PHP con el patrón arquitectónico MVC". Ed. Openlibra. 2011.
- Cockburn, Alistair. "Agile Software Development: The Cooperative Game, Agile Software Development". 2da Edición. Ed. Pearson-Education. 2006.
- Cohn, Mike. "Succeeding with agile software development using scrum". Boston. Ed. Pearson Education. 2010.
- Fowler, Martin. "Patterns of Enterprise Application Architecture". Ed. Addison Wesley, 2002.
- Rumbaugh, James. Ivar Jacobson. Grady Booch. "El proceso unificado de desarrollo de software". 2da Edición. Madrid. Ed. Pearson-Education. 2000.
- Meier, Reto. "Professional Android 2 Application Development". 3ra Edición. Indianapolis. Ed. Wiley. 2010.
- Pressman, Roger S. "Ingeniería del Software, Un Enfoque Práctico". 6ta Edición. Mc Graw Hill, 2005.
- Roberto Rodríguez Echeverría, Encarna Sosa Sánchez, Álvaro Prieto Ramos. "Programación Orientada a Objetos". Ed. Librería Álvaro. 2004.
- Scrum.org. "Scrum.org. Improving the profession of software development". 2013. <http://www.scrum.org/> (accessed 10 7, 2013).
- Sommerville, Ian. "Ingeniería del Software". Madrid. Ed. Pearson-Education. 2005.

Notas Biográficas

MCS. Ulises Ponce Mendoza es profesor de la División de Ingeniería y Tecnologías de la Universidad de la Sierra. Terminó sus estudios de posgrado en Relaciones industriales en el Colegio de Sonora. Participa como instructor certificado de la Academia Local de Networking de CISCO, Egresado de la Lic. en Sistemas Computacionales Administrativos por el Centro de Estudios Superiores del Estado de Sonora.

ITS. Víctor Moreno Yáñez actualmente es desarrollador de software dentro de la empresa eDesarrollos S.A de C.V. Terminó sus estudios de Ingeniería en Telemática y Sistemas en la Universidad de la Sierra.

MCIE. Rafael Soto Bernal actualmente líder y jefe de la empresa eDesarrollos S.A de C.V. Egresado de Ingeniería en Sistemas por el Instituto Tecnológico de Hermosillo, con Maestrías en Consultoría Interna y Externa y Dirección de Organizaciones del Conocimiento.



“2015, Año del Generalísimo José María Morelos y Pavón”.

“2015, Año del Empleo”.

Moctezuma, Sonora, a 17 de marzo de 2015.

MEM-R-002/15.

Asunto: Instrucción respecto presentación de evidencia en informes trimestrales.

LIC. JULIÁN MORENO BARCELÓ

Secretario General Académico

Presente.

Por medio del presente, se le instruye, para que en lo sucesivo, al momento de que personal a su cargo presente los informes trimestrales, se incluya y se anexe evidencia documental en tiempo y forma del cumplimiento de metas del Programa Operativo Anual bajo su responsabilidad, o en caso de que por algún motivo no se cumpla con lo programado, se de la justificación correspondiente.

Lo anterior con el objetivo de que la información sea más clara y transparente para quienes revisan la información emitida por la Institución.

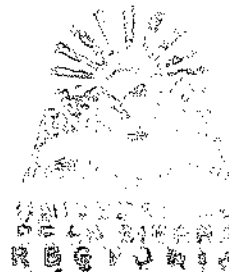
Sin otro particular, aprovecho la ocasión para enviarle un cordial saludo y reiterarme a sus órdenes.

ATENTAMENTE

“Una Puerta Abierta al Cambio y al Desarrollo”

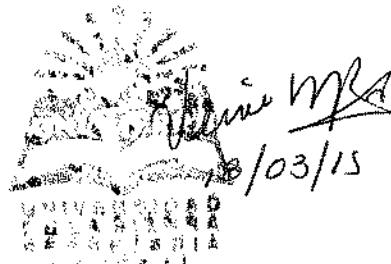
ING. GABRIEL AMAVIZCA HERRERA

Rector.



C.c.p.- Archivo.

GAH/*amvd*



SEC

UNIVERSIDAD DE LA SIERRA | CARRETERA MOCTEZUMA-CUMPAK KM. 2.5 | TEL. +52 (634) 342-96-00
MOCTEZUMA, SONORA, MÉXICO | www.unisierra.edu.mx