

Diseño de una aplicación móvil personalizada de apoyo al aprendizaje de Redes de Computadoras

Matías Loto y Elena Durán

Instituto de Investigaciones en Informática y Sistemas de Información (IISI),
Facultad de Ciencias Exactas y Tecnologías (FCEyT)
Universidad Nacional de Santiago del Estero (UNSE)
matiasunse@gmail.com, eduran@unse.edu.ar

Resumen

Las tecnologías móviles y las comunicaciones están afectando cada vez más la vida de los individuos y son un factor importante a tener en cuenta en la educación. Estas tecnologías han transformado las metodologías de aprendizaje. Hoy el aprendizaje se produce en cualquier momento y en cualquier lugar, y se aprende durante toda la vida. En consecuencia han surgido nuevos modelos tales como: el aprendizaje móvil, el aprendizaje ubicuo, el autoaprendizaje, etc. Es necesario apoyar estos paradigmas educativos con el desarrollo de aplicaciones educativas personalizadas. En este trabajo se presenta el diseño de una aplicación móvil para Tablets y Smartphones con S.O. Android, que utiliza códigos QR para reconocer componentes en un laboratorio de redes de computadoras y presenta información personalizada, de esos componentes, en base al estilo de aprendizaje y nivel de conocimiento del estudiante, asistiéndolo en sus procesos de aprendizaje. La aplicación ha sido diseñada siguiendo la metodología de desarrollo ágil Scrum, caracterizada por el desarrollo iterativo e incremental. La aplicación busca ser una herramienta útil para los estudiantes, por lo que una vez diseñada e implementada, será validada en su entorno real con alumnos de nuestra universidad.

Palabras clave: Computación móvil, estilos de aprendizaje, personalización, códigos QR.

Introducción

El mundo se está transformando a gran velocidad, estamos pasando de una economía industrial a otra basada en la información y orientada por los medios (Cope & Kalantzis, 2009). Un sistema educativo estático y monolítico no se adapta al mundo cambiante, complejo e innovador en el que actualmente vivimos y donde la tecnología cambia de forma dramática y constante. Es necesario adaptarse a este mundo más interconectado, más complejo y que se apoya en la innovación.

Es preciso ajustar los métodos de enseñanza y apoyarlos con la tecnología digital de los tiempos que corren (García, 2009; Mockus et al., 2011). La inclusión de la tecnología y particularmente de la tecnología móvil al proceso educativo es de vital importancia para crear un sistema educativo dinámico que se adapte a un mundo complejo de constante cambio y progreso (García, 2009; Nagella & Govindarajulu, 2008), además como plantea Vahey y Crawford (2002) el aprendizaje con tecnologías móviles puede ser personalizado, situado y auténtico, básicamente más centrado en el estudiante fomentando así la creatividad y la innovación.

La inclusión de la tecnología móvil al área educativa, más que como una opción se presenta como una necesidad para apoyar las tareas de aprendizaje y a su vez motivar a los estudiantes. Considerando esta situación, y con el fin de obtener los mayores beneficios de la tecnología móvil, estamos desarrollando una aplicación educativa personalizada que asista a

los estudiantes en su proceso de aprendizaje en un Laboratorio de Redes de Computadoras. En este trabajo se presenta la concreción de la primera etapa, correspondiente al diseño de dicha aplicación.

En las siguientes secciones se definen algunos conceptos relevantes al trabajo en cuestión; se describe la metodología SCRUM, empleada para el desarrollo de la aplicación; se presenta el Product Backlog con los requisitos de la aplicación, la arquitectura global del sistema y algunos avances respecto a la implementación del prototipo. Finalmente, se presentan las conclusiones del trabajo realizado y las líneas de acción futuras.

Marco Conceptual

Computación Móvil

La Computación Móvil se define como el entorno de computación sobre la movilidad física. En este entorno, los usuarios pueden acceder a datos, información o cualquier otro objeto lógico desde cualquier dispositivo mediante cualquier red mientras se están moviendo (Talukder & Yavagal, 2006; Talukder et al., 2010). También puede definirse como "Un servicio que se mueve con las personas, el cual permite que las mismas se puedan inscribir, recordar, comunicar y razonar independientemente de la posición de los dispositivos." (Lyytinen & Yoo, 2002).

La tecnología móvil posee dos características principales que la hacen diferente a otras formas: movilidad y amplio alcance. Con movilidad nos referimos a la portabilidad basada en el hecho de que los usuarios llevan un dispositivo móvil a todos los lugares a donde se dirigen, pudiendo iniciar el contacto en tiempo real con otros sistemas donde quiera que se encuentren. El amplio alcance hace referencia a la accesibilidad de las personas a las redes inalámbricas que se pueden localizar en cualquier momento. Algunos atributos de la computación móvil son la ubicuidad, comodidad, conectividad instantánea,

localización de productos y servicios y personalización.

Personalización/Adaptación

Incorporar personalización en un sistema de información implica tener en cuenta las necesidades de cada usuario particular y adaptar la presentación de contenido y el comportamiento del sistema en función de las mismas. Para llevar a cabo esta adaptación suelen emplearse un conjunto de modelos abstractos. Tradicionalmente no ha existido un consenso sobre el conjunto de modelos a emplear, pero de forma recurrente aparecen los siguientes modelos en la literatura:

Modelo de dominio: estructura el conocimiento que se desea transmitir y almacena la información por conceptos, las relaciones de éstos con otros conceptos, y sus atributos. Un modelo de dominio o de contenido puede tener un conjunto de conceptos independientes, o puede poseer conceptos relacionados con otros, formando una clase de red semántica. Los sistemas educativos suelen utilizar requisitos previos que están vinculados al concepto; para avanzar a otro concepto se necesita haber conocido una o varias nociones previas (Buitrago, 2010).

Modelo del Usuario: representa la relación de cada sujeto con el conocimiento que se le desea transmitir. Para ello, almacena y estructura aspectos relevantes de cada usuario. "Estos aspectos, llamados atributos, son almacenados en entidades tipo tabla que relacionan a cada usuario con sus características y con los conceptos de Modelo del Dominio" (Buitrago, 2010).

Modelo de Adaptación: ambos tipos de adaptación, la de contenido y la de navegación, ejecutan reglas que especifican qué y cómo se deben mostrar y comportar los elementos del sistema considerando el modelo de usuario.

Metodología SCRUM

Scrum es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. (Schwaber & Sutherland, 2013). Scrum comienza cuando algún interesado necesita un producto.

El Equipo de Scrum incluye 3 roles: El Dueño del Producto, el *Scrum Master*, y los miembros del Equipo de Desarrollo. El Dueño del Producto tiene la responsabilidad de decidir qué trabajo necesita hacerse. El Scrum Master actúa como un líder servicial, ayudando al equipo y a la organización a usar lo mejor posible Scrum. El Equipo de Desarrollo construye el producto incrementalmente, en una serie de periodos de tiempo pequeños llamados *Sprints*. Un Sprint es un periodo de tiempo de duración fija, de una a cuatro semanas, con preferencia por los intervalos más cortos. En cada Sprint, el equipo construye y entrega un Incremento del Producto. Cada incremento es un subconjunto del producto, reconocible y visualmente mejorado, que cumple con el criterio de

aceptación y está construido con un nivel de calidad llamada Definición de Terminado.

Scrum incluye tres artefactos esenciales: el *Product Backlog*, el *Sprint Backlog*, y el Incremento del Producto. El Product Backlog es una lista ordenada de ideas para el producto, ordenada según cómo se espera sean construidas. El Sprint Backlog es un plan detallado para el desarrollo del próximo Sprint. El Incremento del Producto es el resultado requerido de cada Sprint. Es una versión integrada del producto, mantenida con una calidad lo suficientemente alta como para ponerse en producción si el Dueño del Producto lo deseara. Además de estos artefactos, Scrum requiere transparencia dentro del equipo y con los interesados. Por lo tanto, el Equipo de Scrum genera muestras visibles de su planificación y progreso.

Modelado de la Aplicación

Especificación de Requisitos

En la Tabla 1 se presenta una versión resumida de los requisitos o historias de usuario definidas en el Product Backlog.

Tabla 1: Especificación de Requisitos

ID	Nombre	Descripción	Prioridad	Estimación	Solicitante
1	Interfaz de la Aplicación	Se requiere una interfaz de usuario sencilla, fácil de usar pero elegante a la vez.	10	4 semanas	Usuario
2	Login	Se requiere un control de acceso para los estudiantes que ingresan al sistema.	9	4 semanas	Dueño del Producto
3	Leer códigos QR	Se requiere la lectura de códigos QR asociados a puntos de interés (elementos del Laboratorio de Redes de Computadoras sobre los que el alumno puede requerir información)	8	3 semanas	Dueño del Producto
4	Sitio de administración	Se requiere un sitio web que permita fácilmente administrar los parámetros globales, la administración de usuarios y la carga de la base de datos con componentes a ser reconocidos por la aplicación.	8	4 semanas	Dueño del Producto
5	Adaptación	Se requiere un módulo que brinde información personalizada de los puntos de interés en base al modelo de estilo de aprendizaje y nivel de conocimiento del usuario.	8	4 semanas	Dueño del Producto
6	Recuperar contraseñas	Se requiere recuperar la contraseña del usuario.	7	3 semanas	Usuario
7	Dispositivos	Se requiere que la aplicación se adapte a pantallas de Smartphones y Tablets con S.O. Android.	6	3 semanas	Dueño del Producto

Arquitectura del Sistema

Scrum no habla explícitamente de la arquitectura del software, por tal motivo definimos un sprint adicional, llamado “Sprint 0” al inicio del ciclo de desarrollo. El objetivo principal en el Sprint 0 es analizar y diseñar la generalidad del sistema, que satisfaga los requisitos y sea entendible por los miembros del equipo desde sus diferentes puntos de vista durante el desarrollo.

La arquitectura de la aplicación se presenta en la Figura 1. La misma consta de dos grandes componentes: Cliente y Servidor. A grandes rasgos el cliente es la aplicación programada en Android y corre en el dispositivo móvil. El servidor es un sitio web programado en PHP, MySQL y algunos frameworks para administrar todo lo relacionado al usuario, la información sobre los componentes a ser reconocidos y la presentación de la información. Internet es el medio de comunicación entre ambos.

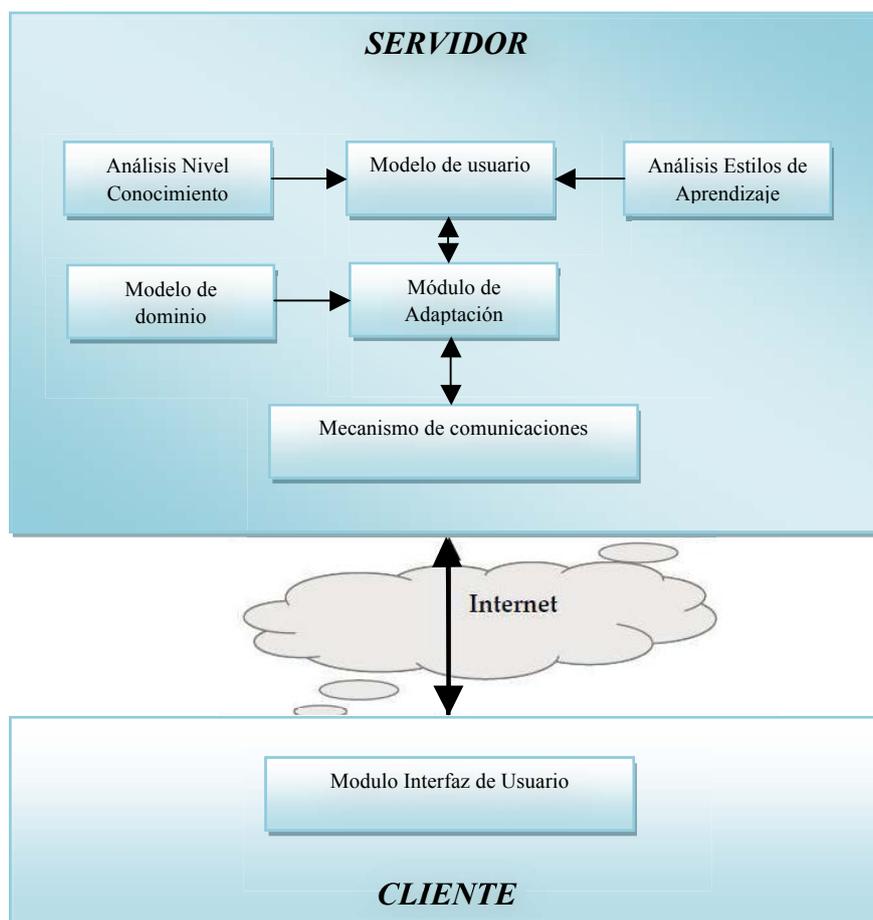


Figura 1: Arquitectura de la Aplicación

El *modelo de usuario* contiene toda la información referida a un usuario (estudiante) y puede determinar su nivel de conocimiento y estilo de aprendizaje en base al *módulo de análisis nivel de conocimiento* encargado de recuperar y actualizar el nivel de conocimiento de un usuario y al *módulo de análisis estilos de aprendizaje* que permite determinar, definir

y actualizar el estilo de aprendizaje de un usuario. Esta información junto con la información del *modelo de dominio*, encargado de recuperar y mantener información actualizada referida a los puntos de interés, son enviados al *módulo de adaptación*, que aplica filtros y permite la personalización de la presentación. Finalmente, ésta información es

transmitida al mecanismo de comunicaciones y recibida por la interfaz de usuario de la aplicación Android en el dispositivo móvil a través de Internet. La adaptación se realiza sobre el contenido y el dispositivo de acceso como se explica en las siguientes secciones.

Hacia el Prototipo

Esta sección presenta la implementación del prototipo inicial basado en la arquitectura planteada.

Adaptación en ambientes de aprendizaje móvil

La adaptación implica considerar cada alumno individualmente y su dispositivo de acceso, que puede ser un Smartphone o una Tablet con Sistema Operativo Android.

- ✓ *Adaptación al dispositivo de acceso*: este tipo de adaptación permite presentar la información al estudiante en smartphones y tablets. Se realiza mediante la API de Android que soporta el desarrollo para múltiples tamaños de pantalla y resoluciones.
- ✓ *Adaptación de Contenido*: esta adaptación se lleva a cabo en base al estilo de aprendizaje y al nivel de conocimiento de cada estudiante. El sistema es capaz de adaptar el contenido que le presenta al usuario (información de los puntos de interés) a los estilos de aprendizaje del modelo de Felder y Silverman (1988), en base a dos tipos de adaptación:
 - *Adaptación al estilo de aprendizaje por defecto*: el sistema inicializa las variables correspondientes del estilo de aprendizaje con valores por defecto sobre la base del índice de estilos de aprendizaje de Felder-Soloman que determina que más estudiantes son activo, sensitivo, secuencial, y visual, en vez de reflexivo, intuitivo, verbal y global (Van Zwanenberg, et al., 2001; Zywno, 2003). Este estilo lo mantiene el sistema hasta

tanto cada estudiante responda el cuestionario de Felder-Silverman.

- *Adaptación al estilo de aprendizaje individual*: el sistema recomienda al estudiante responder el cuestionario de Felder-Silverman; entonces el sistema modifica los valores de las variables del estilo de aprendizaje en base al resultado del cuestionario. Por lo tanto es capaz de adaptar la presentación de contenido al estilo de aprendizaje particular del estudiante.

Los flujos de control en el sistema.

A continuación se describen los principales flujos de control del sistema:

F1: El estímulo que activa el sistema es el usuario cuando inicia la aplicación la cual solicita la registración del mismo. El usuario ingresa sus datos y la aplicación verifica su permiso a través del servidor, si los datos son correctos, se habilita al usuario para ingresar a la pantalla principal de la aplicación.

F2: El usuario puede iniciar el escáner y leer un código QR, correspondiente a la etiqueta de un componente de red, en el Laboratorio de Redes de Computadoras. La información censada por el escáner es enviada al servidor, donde se comprueba si el código corresponde a un punto de interés almacenado en el modelo de dominio.

F3: En base a la identificación de usuario válida, el módulo de adaptación consulta al modelo de usuario sobre el nivel de conocimiento y estilo de aprendizaje del usuario.

F4: En base a la identificación del punto de interés, el módulo de adaptación consulta en el modelo de dominio información acerca del componente de red asociado a esa identificación.

F5: El módulo de adaptación, en base a la aplicación de determinadas reglas, selecciona

el contenido y formato más apropiado para mostrar al estudiante según el estilo de aprendizaje y nivel de conocimiento recuperado, y envía este contenido al cliente para ser desplegado en el dispositivo del usuario.

Clasificación de los estilos de aprendizaje del estudiante y sus reglas de implementación.

Basados en la teoría de estilos de aprendizaje de Felder-Silverman (1988) se define a continuación una clasificación de los estilos de aprendizaje de los estudiantes y las reglas de implementación.

Las dimensiones a considerar en esta primera versión del prototipo son: Sensitivo/Intuitivo y Visual/Verbal; por ser las dimensiones que más relación tienen con la presentación de contenidos, objetivo principal de esta aplicación.

La *dimensión Sensitivo/Intuitivo* está relacionada con el contenido, más concretamente con el tipo de contenido que se presenta. Los estudiantes sensitivos prefieren la presentación de las explicaciones después de los ejemplos y viceversa para los estudiantes intuitivos. Esto implica la necesidad de una relación de pertenencia entre explicaciones y ejemplos, además se deberá enfatizar la información y ejemplos prácticos, concretos y orientados a hechos en el caso de estudiantes sensitivos y por otra parte se hará hincapié en información conceptual, abstracta y teórica para los estudiantes intuitivos.

La *dimensión Visual/Verbal*: está relacionada con la presentación de la información, es una dimensión que puede resultar difícil de adaptar ya que puede eliminar material educativo, se debe ser muy cuidadoso con esto, además se debe tener en cuenta que todo contenido debe poder expresarse de ambas formas, mediante palabras y mediante imágenes (algo difícil de asegurar a priori).

Teniendo en cuenta las dimensiones antes descritas se desprenden cuatro posibles modelos de presentación de información, que se presentan en la Tabla 1.

Tabla 1: Estilos de Aprendizaje Visual y Verbal

	Sensorial	Intuitivo
Visual	1) Ejemplos prácticos y concretos que incluyan gráficos, dibujos, imágenes. 2) Ejemplos orientados a hechos y procedimientos en lo posible con imágenes. 3) Teoría mínima o relevante 4) Sugerir ejercicios prácticos.	1) Teoría, principios, conceptos (con imágenes en lo posible). 2) Ejemplos teóricos y abstractos (con imágenes en lo posible).
Verbal	1) Ejemplos prácticos y concretos con información textual. 2) Teoría mínima y relacionada a hechos. 3) Información escrita o hablada.	1) Teoría, principios, conceptos. 2) Ejemplos teóricos y abstractos. 3) Información escrita o hablada.

Implementación del prototipo

La aplicación se está desarrollando bajo Android y del lado del servidor se utilizará PHP y MySQL.

El soporte para diferentes dimensiones de pantalla se realizara mediante la API de Android que provee los mecanismos necesarios para que la aplicación se adapte y funcione correctamente en las pantallas de Smartphones y Tablets.

A continuación se presentan algunas ilustraciones de una versión inicial del prototipo.

En la Figura 2, se puede observar la interfaz principal de la aplicación en la que se solicita la registración del usuario.

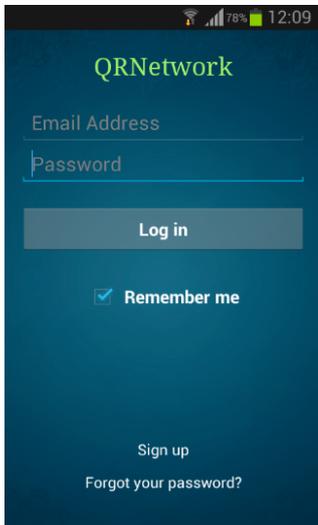


Figura 2: Login

Una vez que el mismo esta logueado, y se encuentra en el Laboratorio de Redes, la aplicación espera por la lectura de un código QR (Figura 3).

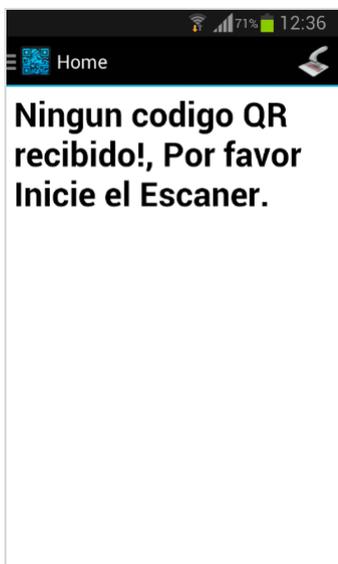


Figura 1: Aplicación a la espera

Una vez que se ha iniciado el escáner, se puede realizar la lectura de un código asociado a un punto de interés. Estos son cada uno de los componentes de red que se encuentran en el Laboratorio y que pueden ser objeto de estudio para el alumno. (Figura 4)



Figura 4: Lectura de un Código QR

A continuación la aplicación hace una solicitud al servidor y éste envía información personalizada del componente reconocido en base a estilo de aprendizaje y nivel de conocimiento del usuario que se encuentra registrado (Figura 5 y 6)



Figura 2: Información del componente



Figura 3: Presentando más detalles

Conclusión

En este artículo presentamos el diseño de una aplicación móvil para asistir el aprendizaje de estudiantes universitarios en un Laboratorio de Redes de Computadoras. Con esta herramienta de software se busca proporcionar un recurso de aprendizaje que asista al estudiante en un proceso de autoaprendizaje mediado por las TIC y aproveche las facilidades de movilidad que proporcionan recursos tales como Tablets y Smartphones, tan usados actualmente por los estudiantes.

Por otra parte, se muestra el diseño de un producto amigable, que se adapta a las características particulares de cada estudiante, como es el nivel de conocimiento respecto al tema y su estilo de aprendizaje

Actualmente estamos desarrollando el módulo de adaptación, el que será implementado en el próximo sprint. Finalmente la aplicación será validada en entorno real con alumnos de carreras de Informática de la FCEyT de la Universidad Nacional de Santiago del Estero.

Referencias

- Buitrago A. (2010). *Sistemas hipermedia Adaptativos (sha) iv*. Universidad de Castilla-La Mancha. España.
- Carver C.A., Howard R.A. & Lane W.D. (1999). *Enhancing Student Learning Through Hypermedia Courseware and Incorporation of Student Learning Styles*, IEEE Transactions on Education, vol. 42, no. 1, 1999, pp. 33-38.
- Cope B. & Kalantzis M. (2009). *Ubiquitous Learning*. University of Illinois Press. 279 págs.
- Felder R.M. (1993). *Reaching the Second Tier: Learning and Teaching Styles in College Science Education*. J. Coll. Sci. Teaching, 23, 5, 286-290.
- Felder R.M. & Silverman L.K. (1988). *Learning styles and teaching styles in engineering education*. Engineering Education, 78 (7):674-681.
- García A.I. (2009). *Códigos QR Flexibles: Un Proyecto con Dispositivos Móviles para el Trabajo de Calentamiento en Educación Física*. EmásF, Revista Digital de Educación Física. ISSN 1989-8304. Año 4, Núm. 23. 2013.
- Lyytinen K. & Yoo Y. (2002). *Issues and challenges in ubiquitous computing: Introduction*. In Communications of ACM, Vol. 45, Nº 12, 2002, pp. 62-65.
- Mockus L., Dawson H., Edel-Malizia S., Shaffer D., Sung A.J. & Swaggerty A. (2011). *The Impact of Mobile Access on Motivation: Distance Education Student Perceptions*. Learning Design at Penn State's World Campus.
- Nagella U.B. & Govindarajulu P. (2008). *Adaptive Approaches to Context Aware Mobile Learning Applications*. International Journal of Computer Science and Security. Vol. 2. Issue 2. Págs. 15-26. India.
- Schwaber K. & Sutherland J. (2013). *The Definitive Guide to Scrum: The Rules of the Game*. The Scrum Guide™.

Talukder A.K. & Yavagal R. (2006). *Mobile Computing: Technology, Applications, and Service Creation*. McGraw-Hill Professional.

Talukder A.K., Ahmed H. & Yavagal R. (2010). *Mobile Computing: Technology, Applications, and Service Creation*. Second Edition. McGraw-Hill Professional.

Vahey P. & Crawford V. (2002). *Palm Education Pioneers Program Final Evaluation Report*. Menlo Park, CA: SRI International.

Van Zwanenberg N., Wilkinson L.J. & Anderson A. (2000). *Felder and Silverman's*

Index of Learning Styles and Honey and Mumford's Learning Styles Questionnaire: How do they compare and do they predict academic performance?. Educational Psychology, Vol. 20 (3), 2000, 365-381.

Zywno M.S. (2003). *A Contribution to Validation of Score Meaning for Felder-Soloman's Index of Learning Styles*, Session 2351, accepted to be presented at the 2003 ASEE Annual Conference and Exposition, Nashville, Tennessee, June 23-25, 2003.