

# **TECNOLOGÍA MÓVIL PARA EVALUAR LA CALIDAD DE LAS HERRAMIENTAS DE DISEÑO RÁPIDO PARA GENERAR SISTEMAS DE INFORMACIÓN**

***Laura Silvia Vargas Pérez***

Instituto Tecnológico de Ciudad Madero

*laura.silvia.vargas@gmail.com*

***Ana María Soto Hernández***

Instituto Tecnológico de Ciudad Madero

*sotohana@yahoo.com*

***Jorge Peralta Escobar***

Instituto Tecnológico de Ciudad Madero

*jperalta3008@gmail.com*

***Agustín Francisco Gutiérrez Tornés***

Universidad Autónoma de Guerrero

*afgutierrez@uagro.mx*

***Edgardo Manuel Felipe Riverón***

Centro de Investigación en Computación C.I.C. del I.P.N.

*edgardo@cic.ipn.mx*

***Carlos Eduardo Uc Rios***

Universidad Autónoma de Campeche

*carlos.uc@unini.edu.mx*

***Vanessa Atenea Vargas Pérez***

Universidad Autónoma de Tamaulipas, FI UAT, Tampico, Tamaulipas

*vanessa.atenea@gmail.com*

## Resumen

En el mundo globalizado se requiere contar con herramientas de software que faciliten el desarrollo de aplicaciones rápidas para generar sistemas de información de calidad. La formación de los ingenieros informáticos debe incluir métodos que permitan su evaluación, utilizando las nuevas tecnologías, como la móvil, por ser la más popular. Se propone una metodología para realizar la evaluación técnica de la calidad de las herramientas de diseño de aplicaciones rápidas (RAD tools) para generar sistemas de información, de cualquier tipo, en ambientes visuales, con la cual es posible seleccionar, de manera eficaz y fácil, las mejores herramientas, utilizando la tecnología móvil (tabletas). Los casos de estudio elegidos para las pruebas a las herramientas RAD son las plataformas visuales comerciales (IDE: Entornos de Desarrollo Integrado) Visual Studio.Net, Eclipse y Net Beans. El proceso de prueba se realizó con usuarios expertos (estudiantes de posgrado y profesores) y con novatos (estudiantes de licenciatura) en sistemas computacionales, en instituciones de educación superior, pero también a organizaciones, empresas u otros usuarios finales que desarrollan aplicaciones informáticas, que lo requieran.

**Palabra(s) clave(s):** Ambiente visual, calidad de software, herramientas RAD, modelo de evaluación técnica.

## 1. Introducción

Para mejorar cualquier *software* se requiere evaluar sus atributos. Para ello se utiliza un conjunto de medidas y métricas significativas, que proporcionan indicadores como estrategia de evaluación técnica de la calidad del producto establecida. Este proceso se implementa con una metodología para valorar los requerimientos específicos que el producto de *software* debe cumplir. Es importante que las mediciones sean fáciles de calcular e interpretar en su ejecución y se eliminen ambigüedades. Algunas veces la definición académica de las características de calidad no permite medirlas directamente, por lo cual se requiere establecer métricas que las correlacionen objetivamente con un producto de *software* [16].

El primer paso en la evaluación de un *software* es la determinación de las propiedades relevantes de calidad a considerar mediante un modelo de calidad específico. Éste identifica las características de calidad y sus interrelaciones con los elementos en que se descompone, para facilitar la evaluación cualitativa y cuantitativa del producto. De acuerdo con Gutiérrez [3, 4], generalmente se presentan en formato de árbol de estructura jerárquica donde aparecen las características en su nivel más alto, las subcaracterísticas en el intermedio y los atributos en el más bajo [7, 8, 9, 10, 11].

En la formación de los estudiantes de ingeniería, para enfrentar los retos globales, es primordial conocer las diferentes metodologías que permitan la evaluación de todo tipo de herramientas, que les ayude a medir y a elevar la calidad en sus productos y procesos; se requiere contar, también, con herramientas de software que faciliten el desarrollo de aplicaciones rápidas para generar sistemas de información estandarizados, de cualquier tipo, y que permitan ser evaluados para determinar su calidad.

### **Tipos de medidas**

El tipo de medición requerido depende del propósito de la evaluación. Si el propósito primario es entender y corregir deficiencias, muchas de las mediciones que se realizan son para monitorear y controlar las mejoras. Si se trata de comparar la calidad de un producto con productos alternativos o contra requerimientos, es importante que la especificación de la evaluación se base en un modelo de calidad preciso, métodos de medición, escalas, y rango de niveles para cada métrica, como la norma ISO/IEC 9241 [10].

A nivel internacional surgen periódicamente herramientas para facilitar la creación de nuevos sistemas de información, lo que implica realizar evaluaciones sobre la calidad de estas herramientas para una selección adecuada en determinado entorno de desarrollo.

Tal es el caso de la evaluación y selección de las herramientas conocidas como RAD (*Rapid Application Development*) que forman parte de un IDE (*Integrated Development Environment* -Entorno o Ambiente de Desarrollo Integrado- que

constituye el marco de trabajo de los programadores de aplicaciones para crear sistemas de información, en general.

### **Objetivo**

La formación de ingenieros informáticos debe incluir métodos que les permitan evaluar aquellas herramientas que van a utilizar para el desarrollo de nuevos sistemas de información o aplicaciones informáticas. Para ello, se propone la utilización de la metodología propia que permite hacer un análisis comparativo de diferentes tipos de herramientas para el desarrollo de aplicaciones en ambientes visuales, mediante una propuesta de métricas de calidad [18].

### **Estado del arte y trabajos relacionados**

Desde hace algunos años se han estado proponiendo diversos tipos de modelos de calidad que se han convertido en estándares: ISO/IEC 25000 [11], ISO/IEC 14598 [9], ISO/IEC 9126 [8], así como las IEEE 1061 [7], IEEE 610 [6]. Estos modelos son muy útiles, ya que proporcionan el manejo de métricas y procesos de evaluación de software, pero son muy genéricos, por lo que deben adaptarse para su utilización práctica.

La gran mayoría de los trabajos asociados con esta temática se enfoca a la evaluación de la calidad en el desarrollo de software, como los de Cochea [1], donde implementó métricas de calidad para establecer la posibilidad de obtener la certificación de calidad en el desarrollo de sus productos de software a medida. Olsina [14] cuya especialidad es la aplicación de métricas para evaluar aplicaciones WEB y E-learning, Piattini [15] su trabajo está enfocado a la aplicación de las métricas para los modelos de Procesos de Negocios, Moreno [13] utiliza las Métricas de Calidad del Software para evaluar las aplicaciones de Gramáticas de Contexto. No se encontraron antecedentes donde se evalúe, de manera directa, las herramientas RAD dentro de los IDE's, lo que se ha encontrado tiene, como se comentó, otros fines diversos, para otras aplicaciones de software, más específicas, pero no se contempla este caso específico, donde se evalúa la calidad en uso de las herramientas RAD para implementar Sistemas

de Información, de cualquier tipo, y para aplicaciones genéricas, de manera muy general. En este trabajo se desarrolló la metodología, utilizando los modelos y estándares mencionados, tomando características, subcaracterísticas y atributos distintos de cada uno [3, 4], además de otras cualidades del Inventario de Medidas de Usabilidad del Software SUMI [5], y desarrollando sus propias métricas de evaluación.

## 2. Desarrollo

Se generó una metodología propia con modelos, procesos, técnicas y herramientas, que permite hacer cuadros comparativos para realizar la evaluación técnica de las herramientas RAD. El modelo propuesto se basa en la fusión: adopción y adaptación de las contribuciones señaladas más arriba, para conformar la arquitectura del modelo. Es necesario precisar que las aplicaciones utilizadas para probar la metodología son productos comerciales por lo que no se dispone de la información concerniente a su código fuente. Lo anterior implica que no se tomaron en cuenta las métricas internas.

### Proceso de evaluación

Las actividades de evaluación en el modelo propuesto son las que se indican en la figura 1, donde se observa que aquellos atributos relacionados con la calidad en uso tiene una importancia singular, ya que es desde el punto de vista del usuario.

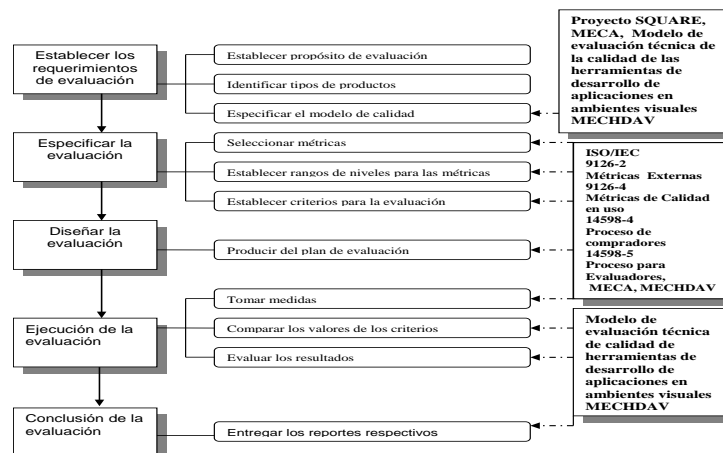


Figura 1 Proceso de evaluación para el modelo propuesto MECHDAV.

La relación de la calidad en uso con las otras características de calidad del software depende del tipo de usuario, como aquel para quien la calidad en uso es principalmente el resultado de la evaluación de la funcionalidad, la confiabilidad, la usabilidad, la eficiencia y la reusabilidad [3, 4, 5].

### Compactación del modelo propuesto

El modelo que se presenta está compuesto por cualidades específicas del *software* (características), subcaracterísticas, atributos y métricas, los cuales se agrupan en dos continentes: el modelo completo y el compacto, para facilitar su enseñanza y comprensión. El uso del modelo completo se recomienda para usuarios expertos, profesores y alumnos de semestres avanzados de ingeniería y posgrado que desarrollan aplicaciones de *software*. Para los usuarios novatos y los alumnos de los primeros semestres de ingeniería se recomienda el modelo compacto (figura 2), *Subconjunto* del modelo MECHDAV. sugerido para la evaluación del producto por un usuario principiante.

#### Característica / Subcaracterística / Atributo / métrica

- 1.3.1.1 **Funcionalidad/Corrección** / Correspondencia de las descripciones con los objetos métrica
- 1.3.3.1 **Funcionalidad/ Corrección** / Utilización correcta del idioma / métrica
- 1.5.1.1 **Funcionalidad/ Normalización** / Normalización de la simbología / métrica
- 1.5.2.1 **Funcionalidad / Normalización**/Normalización del vocabulario / métrica
- 3.1.1.1 **Usabilidad / Aprendibilidad** /Demo / métrica
- 3.1.2.1 **Usabilidad / Aprendibilidad** /Eficacia del Demo / métrica
- 3.1.3.1 **Usabilidad / Aprendibilidad** / Tutorial / métrica
- 3.1.4.1 **Usabilidad / Aprendibilidad** /Eficacia del Tutorial / métrica
- 3.1.5.1 **Usabilidad / Aprendibilidad** /Documentación / métrica
- 3.2.1.1 **Usabilidad / Atractividad** /Interacción atractiva / métrica
- 3.3.1.1 **Usabilidad / Comprensibilidad** /Ayuda en línea / métrica
- 3.3.2.1 **Usabilidad / Comprensibilidad** /Interfaz de usuario adecuada / métrica
- 3.3.3.1 **Usabilidad / Comprensibilidad** /Terminología acorde al usuario / métrica
- 3.4.1.1 **Usabilidad / Difusión** /Amplitud / métrica
- 3.4.2.1 **Usabilidad / Difusión** /Frecuencia de operación / métrica
- 3.5.1.1 **Usabilidad / Operabilidad**/Operabilidad de las ayudas / métrica
- 3.5.2.1 **Usabilidad / Operabilidad** /Utilidad de las ayudas / métrica
- 5.1.1.1 **Portabilidad / Instalabilidad** /Módulo de instalación / métrica
- 5.1.2.1 **Portabilidad / Instalabilidad** /Documentación del módulo de instalación / métrica
- 5.1.3.1 **Portabilidad / Instalabilidad** /Módulo de configuración / métrica
- 5.1.4.1 **Portabilidad / Instalabilidad** /Documentación del módulo de configuración / métrica
- 6.3.1.1 **Calidad en uso / Satisfacción** / efectos psicológicos favoritos de usuario: SUMI métricas

Figura 2 *Subconjunto* del modelo MECHDAV.

En la figura 3 se muestra el modelo MECHDAV completo, ya que este modelo es el que contiene todos los elementos requeridos para evaluar las herramientas RAD, y las métricas con las que se efectúan los cálculos que determinan la calificación de la calidad de estas herramientas. Aunque uno de los propósitos de la aplicación de este modelo de evaluación técnica es la de proporcionar un medio de comparación objetivo entre diversos productos de *software* dentro de un ambiente visual para cualquier tipo de usuario (sea experto o principiante).

### Características /Subcaracterísticas / Atributo / Métrica

- 1.1.1.1 Funcionalidad / **Complejidad** / Contenido total / métrica
- 1.2.1.1 Funcionalidad / **Consistencia** / Uniformidad del formato de los elementos funcionales / métrica
- 1.2.2.1 Funcionalidad/ **Consistencia** / Uniformidad de retorno al procesamiento / métrica
- 1.2.3.1 Funcionalidad/ **Consistencia** /Uniformidad del vocabulario y de la simbología utilizada / métrica
- 1.3.1.1 Funcionalidad/ **Corrección** / Correspondencia de las descripciones con los objetos / métrica
- 1.3.2.1 Funcionalidad/ **Corrección** / Funcionamiento correcto / métrica
- 1.3.3.1 Funcionalidad/ **Corrección** / Utilización correcta del idioma / métrica
- 1.4.1.1 Funcionalidad/ **Interoperabilidad** / Intercambiabilidad de datos / métrica
- 1.4.2.1 Funcionalidad/ **Interoperabilidad**/ Intercambiabilidad en componentes e interfaces / métrica
- 1.5.1.1 Funcionalidad/ **Normalización** / Normalización de la simbología / métrica
- 1.5.2.1 Funcionalidad / **Normalización**/ Normalización del vocabulario / métrica
- 2.1.1.1 Confiabilidad / **Madurez** / Frecuencia de fallas / métrica
- 2.2.1.1 Confiabilidad / **Recuperabilidad** / Opciones de recuperabilidad / métrica
- 2.3.1.1 Confiabilidad / **Tolerancia de errores o fallos** /Procesamiento degradado / métrica
- 2.3.2.1 Confiabilidad/ **Tolerancia de errores o fallos** / Tratamiento de errores / métrica
- 3.1.1.1 Usabilidad / **Aprendibilidad** / Demo / métrica
- 3.1.2.1 Usabilidad / **Aprendibilidad** / Eficacia del Demo / métrica
- 3.1.3.1 Usabilidad / **Aprendibilidad** / Tutorial / métrica
- 3.1.4.1 Usabilidad / **Aprendibilidad** / Eficacia del Tutorial / métrica
- 3.1.5.1 Usabilidad / **Aprendibilidad** / Documentación / métrica
- 3.2.1.1 Usabilidad / **Atractividad** / Interacción atractiva / métrica
- 3.2.2.1 Usabilidad / **Atractividad** / Recuperación exitosa / métrica
- 3.2.3.1 Usabilidad / **Atractividad** / Tiempo de operación / métrica
- 3.3.1.1 Usabilidad / **Comprensibilidad** / Ayuda en línea / métrica
- 3.3.2.1 Usabilidad / **Comprensibilidad** / Interfaz de usuario adecuada / métrica
- 3.3.3.1 Usabilidad / **Comprensibilidad** / Terminología acorde al usuario / métrica
- 3.4.1.1 Usabilidad / **Difusión** / Amplitud / métrica
- 3.4.2.1 Usabilidad / **Difusión** / Frecuencia de operación / métrica
- 3.5.1.1 Usabilidad / **Operabilidad**/ Operabilidad de las ayudas / métricas
- 3.5.2.1 Usabilidad / **Operabilidad** / Utilidad de las ayudas / métrica
- 4.1.1.1 Eficiencia / **Escalabilidad** / Disponibilidad / métrica.
- 4.2.1.1 Eficiencia / **Uso de los recursos** / Eficiencia en los recursos / métrica
- 4.3.1.1 Eficiencia / **Uso del tiempo** / Eficiencia en tiempo / métrica
- 5.1.1.1 Portabilidad / **Instalabilidad** / Módulo de instalación / métrica
- 5.1.2.1 Portabilidad / **Instalabilidad** / Documentación del módulo de instalación / métrica
- 5.1.3.1 Portabilidad / **Instalabilidad** / Módulo de configuración / métrica.
- 5.1.4.1 Portabilidad / **Instalabilidad** / Documentación del módulo de configuración / métrica.
- 5.2.1.1 Portabilidad / **Adaptabilidad** / Independencia del hardware / métrica.
- 5.2.2.1 Portabilidad / **Adaptabilidad** / Independencia del ambiente de software / métrica.
- 6.1.1.1 Calidad en uso / **Eficacia** / Eficacia de tareas / métrica.
- 6.1.2.1 Calidad en uso / **Eficacia** / Rendimiento de tareas / métrica.
- 6.2.1.1 Calidad en uso / **Productividad** / Eficiencia relativa al usuario / métrica.
- 6.2.2.1 Calidad en uso / **Productividad** / Proporción productiva / métrica.
- 6.3.1.1 Calidad en uso / **Satisfacción** / efectos psicológicos favoritos de usuario: SUMI métricas.

Figura 3 Modelo MECHDAV completo.

## Definición de las métricas y de la escala evaluativa

Las evaluaciones se realizan midiendo los atributos a los que se les asigna una métrica correspondiente. El resultado, el valor medido, puede ser *mapeado* sobre una escala. Este valor no muestra en sí mismo el nivel de satisfacción de los requerimientos. Para este propósito la escala se divide en rangos que definen el grado de satisfacción.

El modelo utiliza dos maneras para hacerlo:

- Dividir la escala en dos categorías como son insatisfactorio y satisfactorio.
- Dividir la escala en cinco niveles o categorías denominadas A, B, C, D y E.

Observar la figura 4. Dado que en la norma IEEE 610 [6] se define la métrica como “una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado”, para usar la segunda forma se propone un conjunto de rangos de escala numérica donde se califica cada una de las métricas.

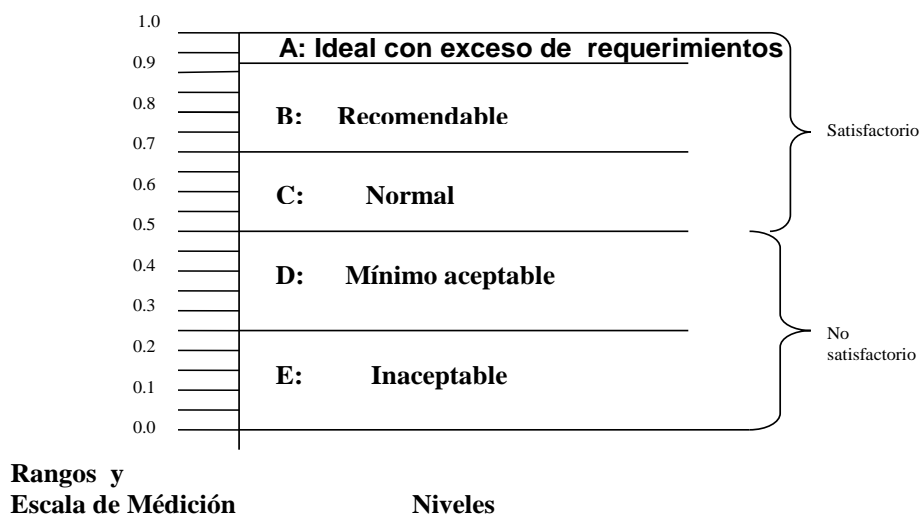


Figura 4 Rangos y escala de medición para las métricas.

La evaluación se realiza llenando un cuestionario donde, al seleccionar una respuesta, se determina el valor de la métrica, asociada a la escala presentada en la tabla 1. Para respaldar este modelo se han desarrollado y documentado 44 métricas, tal como aparece en el formato de la figura 5, más 11 métricas más, que corresponden a la característica: Calidad en Uso; subcaracterística: Satisfacción; atributo: Efectos psicológicos de usuario.



Tabla 1. Interpretación de rangos de niveles de métricas.

VALOR	% CUMPLIMIENTO	SIGNIFICADO / INTERPRETACIÓN	RANGO
1.0	90 - 100	Excelente / Siempre	A
0.8	70 - 89	Satisfactorio / Casi siempre	B
0.6	50 - 69	Aceptable / Regularmente	C
0.4	30 - 49	Deficiente / En ocasiones	D
0.0	0 - 29	Inaceptable / Nunca o raras veces	E

**Característica: 3. USABILIDAD.**

**Subcaracterística: 3.5 Operabilidad.**

**Atributo: 3.5.2 Operabilidad de las ayudas.**

**Métrica: 3.5.2.1 Relación de resultados exitosos mediante la facilidad de operación de las ayudas del software.**

**Objetivo:** Determinar la relación del esfuerzo realizado por el usuario para obtener primeros resultados exitosos mediante la facilidad de operación de las ayudas del software.

**Método:** Analizar los componentes de cada herramienta para determinar la relación de resultados exitosos mediante la facilidad de operación de las ayudas del software.

**Fórmula:**  $X = A/B$

**Evaluación:**  $E(x) = \{ (0,0), (0.4, 40), (0.6, 60), (0.8, 80), (1,100) \}$

**Medidas:** A= Número de primeras tareas cuyos objetivos son completados exitosamente, utilizando las ayudas.

B= Número de tareas intentadas por el usuario.

**Interpretación:** Guía interactiva (intentos de usuario en tareas exitosas)

$0 < X <= 1$ ; lo más cercano a 1 es lo mejor.

**Fuente de referencia:** nueva de MECHDAV, ISO 9126.

Figura 5 Ejemplo de la documentación de una de las métricas utilizadas.

Las descripciones y aplicaciones de las métricas se enfocan en: la observación del comportamiento del *software* en cuanto a la diferencia entre los resultados de ejecución actuales y la especificación de los requerimientos (una vista de prueba y validación de calidad); las ocurrencias inesperadas del comportamiento en el tiempo o utilización de recursos durante la operación del *software*; las instrucciones directas al usuario para que realice una tarea específica, tomando nota de ciertos indicadores (por ejemplo: tiempo, número de ocurrencias de algún evento, etc.); y los cuestionamientos directos al usuario, en los cuales sólo se determina la existencia (presencia o ausencia) de la herramienta y/o la característica a analizar (atributos esenciales).

Y en la figura 6 se muestra la sistematización de la métrica mostrada en la figura 5.

	A:	B:	X:
1) Espacios de trabajo:	8	10	0.8
2) Asistentes:	6	10	0.6
3) Editores de recursos:	6	10	0.6
4) Paneles de vistas jerárquicos:	8	10	0.8
5) Interprete de línea de comandos:	8	10	0.8
6) Biblioteca de clases base:	8	10	0.8
7) Acceso directo al API:	10	10	1
8) Herramientas de seguridad:	8	10	0.8
9) Generador de componentes:	8	10	0.8
10) Incorporación de componentes avanzados:	10	10	1
11) Servicios WEB:	8	10	0.8

Figura 6 Muestra la sistematización de la métrica de la figura 5 dentro de la herramienta.

### 3. Resultados

Plasmar los resultados tanto parciales como totales de la evaluación de la calidad del *software* no es tarea fácil, por lo que se deben elegir formatos simples y comprensibles para conseguir una valoración rápida y confiable de la calidad de sus diferentes representaciones. Los casos de estudio elegidos para las pruebas de las herramientas RAD son las plataformas visuales comerciales Visual Studio.Net [12], Net Beans [17] y Eclipse [2] , en diferentes versiones. Estos paquetes de software no requieren la compra de las licencias, porque dos son productos de código abierto (Net Beans y Eclipse), son gratuitos y sin restricciones de uso; el único producto que es de licencia comercial para su venta (Visual Studio.Net), es otorgado a las instituciones educativas, para uso de sus estudiantes y docentes, de manera gratuita, durante el periodo escolar.

#### Prototipo de la herramienta de Software

El prototipo de PROYEVA fue desarrollado en ambiente WEB responsivo, con lenguajes de programación HTML5, CSS3 Javascript y PHP, además de un manejador de base de datos en MySQL que se instaló en un servidor HP Proliant DL380 con un sistema operativo de red Windows Server 2008, la red interna que se utilizó fue WIFI, y la recepción con 10 tabletas de 7" marca Lenovo y con un sistema operativo Android 4.2.

Este sistema fue probado y usado, con mucha aceptación, por estudiantes de licenciatura y de postgrado, quienes hicieron diversas pruebas a los paquetes de software mencionados, en los dispositivos móviles disponibles (tabletas). La tecnología móvil es lo que más utilizan los estudiantes, por lo práctica, moderna, económica, sencilla, llamativa y útil para realizar sus tareas escolares de diversos tipos, y se les genera una nueva aplicación para que ellos determinen la calidad de las diversas herramientas RAD con las que pueden construir sus sistemas de información, eligiendo las que muestren mayor puntaje en su calidad. Los resultados obtenidos por la evaluación técnica durante todo el proceso de medición, para cada una de las herramientas RAD -Visual Studio.Net, Net Beans y Eclipse- se muestran en las figuras 7, 8, 9, 10, 11 y 12. Para este fin se eligieron formatos tales como listas de comprobación (*checklist*), tablas simples de relación y matrices de control.

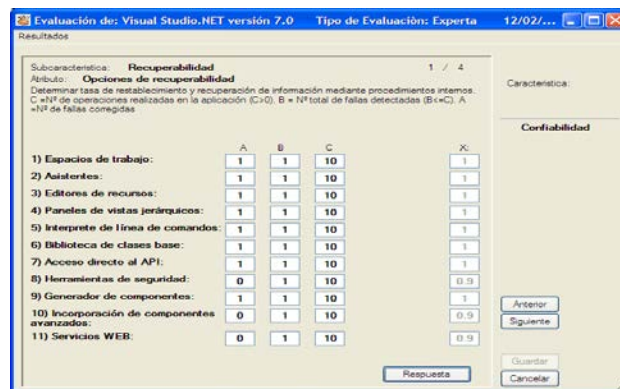


Figura 7 Tablas de relación, valores ingresados se operan para obtener métrica atributo.

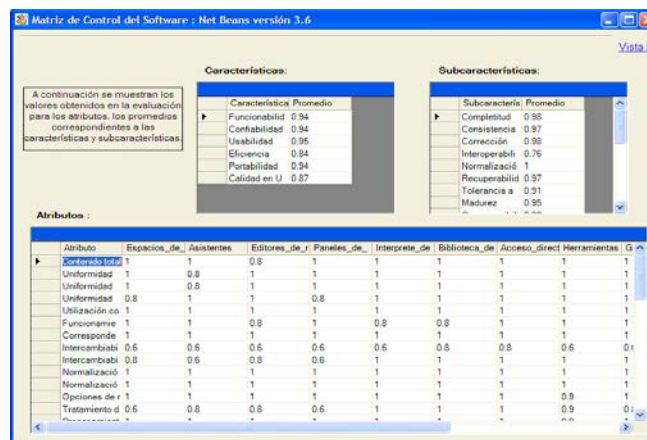


Figura 8 Matriz de control que se genera con evaluación del ambiente Visual Net Beans.

**Reporte de Evaluación del Software : Visual Studio.NET ver. 2003**

**MECHDAV**

Tipo de Evaluación: Experta  
 Nombre del Ambiente: Visual Studio.NET ver. 2003  
 Descripción: Ambiente visual utilizado en el desarrollo de aplicaciones creado por la e

Características	Promedio	Nivel de Clasificación	Criterio	Conclusiones
Funcionalidad	0.95	<input type="checkbox"/> Excelente	<input checked="" type="checkbox"/> Sin modificación	<input checked="" type="checkbox"/> Aceptado
Confiabilidad	0.92	<input checked="" type="checkbox"/> Satisfactorio	<input type="checkbox"/> Pequeñas modificaciones	<input type="checkbox"/> Rechazado
Usabilidad	0.88	<input type="checkbox"/> Aceptable	<input type="checkbox"/> Grandes modificaciones	
Eficiencia	0.87	<input type="checkbox"/> Deficiente		
Portabilidad	0.82	<input type="checkbox"/> Inaceptable		
Calidad en Uso	0.9			
<b>TOTAL:</b>	<b>0.89</b>			

Nombre del Evaluador: Ing. Laura Silvia Vargas Pérez M.C.  
 Organización: Instituto Tecnológico de Ciudad Madero  
 Cargo: Profesor Titular de la Licenciatura en Informática e Ingeniería en Sistemas Computacionales  
 Área: Departamento de Sistemas y Computación

Salir

Figura 9 Reporte técnico final de evaluación del ambiente Visual Studio.Net, versión 2003.

**Report of Software Evaluation: Eclipse 3.1.1 - Evaluation Type: Expert - Date: Tuesday, 02-17-20...**

**MECRAD**

Type of Evaluation: Expert  
 Environment Name: Eclipse 3.1.1  
 Description: Visual Environment used for development Applications

Features	Average	Classification Level	Criterion	Conclusions
Functionality	0.96	<input checked="" type="checkbox"/> Excellent	<input checked="" type="checkbox"/> Without Modification	<input checked="" type="checkbox"/> Accepted
Reliability	0.96	<input type="checkbox"/> Satisfactory	<input type="checkbox"/> Little Modifications	<input type="checkbox"/> Rejected
Usability	0.93	<input type="checkbox"/> Acceptable	<input type="checkbox"/> Big Modifications	
Efficiency	0.91	<input type="checkbox"/> Deficient		
Portability	0.97	<input type="checkbox"/> Unacceptable		
Quality in Use	0.82			
<b>TOTAL:</b>	<b>0.93</b>			

Evaluator Name: Ing. Jorge Peralta Escobar M.S.I.  
 Organization: Technological Institute of Madero City Autonomous Tamaulipas University  
 Position: Computer Science Professor  
 Area: Department of Computer Science

Exit

Figura 10 Reporte técnico final de evaluación de sistema del ambiente Eclipse (posgrado).

**Report of Software Evaluation: Netbeans 8.0 - Evaluation Type: Expert - Date: Tuesday, 10-21-2014**

**MECRAD**

Type of Evaluation: Expert  
 Environment Name: Netbeans 8.0  
 Description: Visual Environment used for development Applications  
 Date on which was the evaluation: Tuesday, 10-21-2014

Features	Average	Classification Level	Criterion	Conclusions
Functionality	0.94	<input checked="" type="checkbox"/> Excellent	<input checked="" type="checkbox"/> Without Modification	<input checked="" type="checkbox"/> Accepted
Reliability	0.94	<input type="checkbox"/> Satisfactory	<input type="checkbox"/> Little Modifications	<input type="checkbox"/> Rejected
Usability	0.95	<input type="checkbox"/> Acceptable	<input type="checkbox"/> Big Modifications	
Efficiency	0.84	<input type="checkbox"/> Deficient		
Portability	0.94	<input type="checkbox"/> Unacceptable		
Quality in Use	0.87			
<b>TOTAL:</b>	<b>0.91</b>			

Evaluator Name: José Agustín Hoz Ruiz y Jessica Solís Franco  
 Organization: Technological Institute of Madero City  
 Position: Students  
 Area: Department of Computer Science

Exit

Figura 11 Reporte técnico final de evaluación de sistema, estudiante posgrado.

Report of Software Evaluation: NetBeans 6.1 Evaluation Type: Expert. Date: Thursday, 05-28-2009

**MECRAD**

Type of Evaluation: Expert  
Environment Name: NetBeans 6.1  
Description: Ambiente de Desarrollo de Aplicaciones  
Date on which was the evaluation: Thursday, 05-28-2009

Features	Average	Classification Level	Criterion	Conclusions
Functionality	0.90	<input type="checkbox"/> Excellent	<input checked="" type="checkbox"/> Without Modification	<input checked="" type="checkbox"/> Accepted
Reliability	0.90	<input checked="" type="checkbox"/> Satisfactory	<input type="checkbox"/> Little Modifications	<input type="checkbox"/> Rejected
Usability	0.80	<input type="checkbox"/> Acceptable	<input type="checkbox"/> Big Modifications	
Efficiency	0.80	<input type="checkbox"/> Deficient		
Portability	0.80	<input type="checkbox"/> Unacceptable		
Quality in Use	0.80			
TOTAL:		<b>0.83</b>		

Evaluator Name: Hector Rodrigo Ramirez Sánchez  
Organization: ITCM  
Position: Estudiante  
Area: Dpto. Sistemas y Computación

Exit

Figura 12 Reporte técnico final de evaluación del sistema, estudiante del ISC.

Se realizaron diez pruebas para los ambientes visuales estudiados. Seis de ellos se realizaron por usuarios expertos (maestros y alumnos del posgrado del ITCM) y cuatro por principiantes (alumnos de ingeniería en sistemas computacionales del ITCM). Las evaluaciones las realizaron los alumnos de la licenciatura de Ingeniería en Sistemas Computacionales (usuarios inexpertos), profesores y alumnos del posgrado de la Maestría en Ciencias de la Computación (usuarios expertos).

Cuando se obtienen los valores respectivos de la evaluación del *software* elegido, se genera el reporte final de la evaluación, en el que se plasman los resultados definitivos y el porcentaje de cumplimiento. A su vez, se proporciona un esquema donde se muestran los puntos donde el *software* obtiene un buen nivel de calificación de calidad, con qué criterio lo alcanza y además se brinda una recomendación final acerca de su aceptación o rechazo.

#### 4. Discusión

Este proyecto se encuentra terminado en sus primeras fases, que cubre el modelo completo y su metodología para la evaluación técnica de la calidad de las herramientas de desarrollo de aplicaciones rápidas para generar sistemas de información de calidad y con sustentabilidad. Se han desarrollado los primeros prototipos del software, que es la herramienta operando en máquinas de escritorio

PC; en esta propuesta, ahora se está trabajando para que opere en dispositivos móviles.

De acuerdo con los resultados obtenidos en las pruebas realizadas a los ambientes de desarrollo integrado (IDE) y sus herramientas RAD, seleccionados como casos de estudio, se llegó a las conclusiones que se presentan a continuación:

El Visual Studio.Net (Microsoft, 2006) obtuvo una evaluación general promedio de 0.89 (89%) entre los usuarios principiantes y un puntaje de 0.88 (88%) entre los expertos, dentro del rango de valores propuesto, por lo que el nivel de clasificación de calidad alcanzado es satisfactorio sin recomendaciones, ya que no requiere de modificaciones en su diseño -sólo actualizaciones- y por lo tanto es aceptado ampliamente. Vale la pena comentar que una de sus mayores debilidades se ubica en la característica de portabilidad, lo cual es comprensible dada su dependencia de la plataforma Windows de Microsoft (figura 9).

Eclipse (Eclipse Foundation, 2005) y Net Beans (Sun Microsystems, 2004-2010) obtuvieron una evaluación general promedio de 0.91 (91%) y 0.93 (93%), respectivamente, entre los usuarios expertos que lo evaluaron (figuras 10 y 11). Y el dictamen del informe del Reporte técnico final define que el nivel de clasificación de calidad alcanzado es excelente, sin recomendaciones, en ambos casos, ya que no requiere de modificaciones en su diseño (sólo actualizaciones) por lo que se recomienda ampliamente; aunque se puede comentar que se podría pensar que su mayor debilidad se ubica en su calidad en uso, dado que para la utilización de ambos es requisito indispensable el dominio del lenguaje Java, pero como se evalúan las herramientas RAD, no influye en el resultado.

Esta herramienta le permitirá al usuario final que realiza las pruebas, conocer los dictámenes de la calidad en uso que le ofrecen las diferentes herramientas RAD de cada plataforma comercial (IDE), en sus diversas versiones, para que tome la decisión de que IDE y que herramientas le convendrá utilizar para desarrollar sus aplicaciones informáticas diversas; el usuario podrá seleccionar las que considere convenientes para generar sus sistemas de información, con base en su calidad en uso, su utilidad, experiencia y otros factores.

## **5. Conclusiones**

La tecnología móvil es la más utilizada por los estudiantes, debido a su bajo costo, facilidad de uso, una gama amplia de aplicaciones que puede obtener gratuitamente de internet, etc. Es por esto que se genera esta aplicación para que pueda operar, también, en dispositivos móviles (celulares y tabletas), así como en la internet, y en la modalidad monousuario.

El modelo y la metodología propuesta tiene cuatro niveles de calidad, donde el último son las medidas-métricas (que fueron creadas específicamente para este modelo), en las cuales se definen los cálculos de la calidad en uso de cada una de las herramientas RAD, de acuerdo a sus características más genéricas, que conforman al IDE en proceso de evaluación, para generar cualquier aplicación en los sistemas de información a crear, y eligiendo a las mejores, dentro del IDE's de mayor calidad, de aumentará la posibilidad de generar los sistemas de información y sus aplicaciones de mayor calidad.

Debido a la similitud de los resultados de la evaluación técnica, en las plataformas utilizadas como casos de estudio, para decidir cuál seleccionar, se deben tomar en cuenta otros criterios, también importantes, como son los costos, la posibilidad de interacción con otros sistemas, la posible inserción en la plataforma en la que se van a desarrollar los sistemas, etc.

Como trabajo futuro se recomienda hacer revisiones periódicas al modelo y a la metodología para su actualización, seguir estudiando las herramientas de desarrollo de aplicaciones rápidas, para incorporar otras al modelo de evaluación, sustituir las que hayan evolucionado y que ya no sean del mismo tipo de herramientas que generen aplicaciones rápidas para los sistemas de información. donde las métricas se pueden ir adaptando para nuevas herramientas RAD y para nuevas características que vayan surgiendo dentro de los IDE's, que también están evolucionando con mucha rapidez en el mercado (las libres y de licencia). Se puede instalar en ambientes multiusuario (producto terminado de software), en un ambiente WEB (arquitectura cliente servidor), y realizar las pruebas pertinentes de cada caso. Funciona perfectamente en cualquier dispositivo móvil (tabletas, celulares), así como tratar de introducir el modelo de evaluación técnica de las

herramientas de desarrollo de aplicaciones en los ambientes visuales de los sitios *web* y en ambientes de otro tipo.

## **6. Bibliografía y Referencias**

- [1] T. Cochea, Métrica de Calidad de los sistemas de información – aplicación en la certificación de calidad de un Sistema de una empresa del sector hidrocarburífero. 2009. Artículos de Tesis de Grado. ICM, Escuela Superior Politécnica del Litoral. Guayaquil, Ecuador.
- [2] Eclipse Foundation. <http://www.eclipse.org>.
- [3] Gutiérrez, Modelo de evaluación para el aseguramiento de la calidad del Software, Modelo MECA. 2003. Instituto Politécnico Nacional. México.
- [4] Human Factors Research Group. SUMI: Software Usability Measurement Inventory, European Directive on Minimum Health and Safety Requirements for Work with Display Screen Equipment. Londres: Taylor & Francis Ltd. (2000).
- [5] IEEE. Software Engineering Standards Collection. Standard Glossary of Software Engineering Terminology. IEEE, Std 610. 1994. Institute of Electrical and Electronics Engineers. Estados Unidos de América.
- [6] IEEE. Software Quality Metrics Methodology. IEEE 1061 Standard for a Software Quality Metrics Methodology. 1992. Institute of Electrical and Electronics Engineers. Estados Unidos de América.
- [7] ISO. ISO/IEC 9126. Software Product Evaluation – Part 2: External metrics. Part 4: Quality in use metrics--. International Organization for Standardization. Suiza. 1997.
- [8] ISO. ISO/IEC 14598. Information technology -- Software product evaluation – Part 4: Process for evaluators, Part 5: Process for acquirers. International Organization for Standardization. Suiza. (1998).
- [9] ISO. ISO 9241-151. Ergonomics of human-system interaction -- Part 151: Guidance on World Wide Web user interfaces and industry standards for user centered design. International Organization for Standardization. (2008). Suiza.



- [10] ISO. *ISO/IEC 25000 Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE)—Guide to SQuaRE*. International Organization for Standardization. (2005). Suiza.
- [11] Microsoft. *Visual Studio.Net 2003*. Entorno de Desarrollo Integrado IDE para sistemas operativos Windows. Versión 7.1. <https://www.visualstudio.com/es-es/visual-studio-homepage-vs.aspx>.
- [12] M. Moreno. “Aplicación de las Métricas de Calidad del Software en la Evaluación Objetiva de Gramáticas Independientes de Contexto Inferidas”. I Simposio Avances en Gestión de Proyectos y Calidad del Software. Universidad de Salamanca. 2004. Pp. 209-220.
- [13] L. Olsina, “Medición y Evaluación de Calidad en Uso: Un Caso de Estudio para una Aplicación E-Learning”. 9º Taller Iberoamericano de Ingeniería de Requisitos y Ambientes de Software. La Plata: Centro Latinoamericano de Estudios en Informática. 2006. Pp. 317-33.
- [14] M. Piattini, “Métricas para la Evaluación de Modelos de Proceso de Negocio”. 9º Taller Iberoamericano de Ingeniería de Requisitos y Ambientes de Software. 2006. Pp. 419-432.
- [15] R. Pressman, *Ingeniería de Software. Un enfoque práctico*. Séptima Edición. 2010. McGraw Hill/Interamericana. Madrid, España.
- [16] Sun Microsystems *NetBeans Entorno de Desarrollo Integrado IDE libre de código abierto*. Versión 3.6., Versión 8.0. Sun Microsystems / Oracle Corporation. <https://netbeans.org/>.
- [17] L. S. Vargas-Pérez. A. F. Gutiérrez-Tornés, E. M. Felipe- Riverón, “MECRAD: Model and Tool for the Technical Quality Evaluation of Software Products in Visual Environment”. 3rd International Multi-Conference on Computing in the Global Information Technology. 2008. Pp. 107-112.
- [18] L. S. Vargas-Pérez, A. F. Gutiérrez-Tornés, E. M. Felipe- Riverón. *Sistema para evaluar herramientas RAD*. Registro ante SEP INDAUTOR con número de registro público 03-2012-100211292400-01. 2012. Ciudad de México, México.

## **7. Autores**

M.C. Laura Silvia Vargas Pérez, Ingeniera Electrónica, Maestra en Ciencias en Computación Electrónica por el Centro de Investigación en Computación del Instituto Politécnico Nacional.

Dr. Agustín Francisco Gutiérrez Tornés, Doctor en Ciencias PhD por la Universidad de Varsovia, Polonia.

Dr. Edgardo Manuel Felipe Riverón, PhD por el Instituto de Investigación en Computación y Automatización de la Universidad de Budapest, Hungría.

Dra. Ana María Soto Hernández, Maestra en Ciencias en Matemática Educativa por el Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Doctora en Educación por la Universidad Autónoma de Tamaulipas.

Dr. Carlos Eduardo Uc Ríos, Doctor en Ciencias del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional.

M.S.I. Jorge Peralta Escobar, Maestro en Sistemas de Información por la Universidad Autónoma de Tamaulipas.

M.A.I. Vanessa Atenea Vargas Pérez, Maestra en Administración Industrial por la Universidad Autónoma de Tamaulipas, Maestra en Finanzas y Economía Internacional por la Universidad de Barcelona, España.