



# Integración de pruebas remotas de usabilidad en Programación Extrema: revisión de literatura

*Integration of Remote Usability Tests in eXtreme Programming: A Literature Review*

*Integração de Testes de Usabilidade Remotos em Programação Extrema: Revisão de Documentos*

**Diego Sánchez-Hernández**

[diegosh18@hotmail.com](mailto:diegosh18@hotmail.com)

Escuela de Informática

Universidad Nacional

Heredia, Costa Rica.

Orcid: <https://orcid.org/0000-0002-8209-077X>

**Fulvio Lizano-Madriz**

[fulvio.lizano.madriz@una.cr](mailto:fulvio.lizano.madriz@una.cr)

Escuela de Informática

Universidad Nacional

Heredia, Costa Rica.

Orcid: <http://orcid.org/0000-0001-8789-2985>

**María Marta Sandoval-Carvajal**

[maria.sandoval.carvajal@una.cr](mailto:maria.sandoval.carvajal@una.cr)

Escuela de Informática

Universidad Nacional

Heredia, Costa Rica.

Orcid: <https://orcid.org/0000-0002-0718-9243>

Recibido-Received: **5/jun/2018** • Corregido-Corrected: **16/oct/2018**.

Aceptado-Accepted: **8/feb/2019** • Publicado-Published: **31/ene/2020**.

## Resumen

El objetivo de este trabajo fue contestar la pregunta de investigación: ¿Cómo se integran las pruebas de usabilidad remota con Programación eXtrema? La metodología utilizada fue la revisión de literatura, a través del proceso de búsqueda, selección y estructuración. Se utilizó palabras clave, categorías emergentes y cadenas de búsquedas a través de reconocidas bases de datos. Como resultado, un análisis del estado actual de los artefactos y procesos que se han utilizado en esta integración, lo que ha llevado a un enfoque combinado en áreas como: 1. Gestión de requerimientos, 2. Manejo de integrantes del equipo, que incluye desarrolladores, evaluadores de usabilidad, usuarios expertos y finales, 3. Manejo de pruebas, datos recolectados y retroalimentación, 4. Diseño y 5. Las actividades que se desarrollan dentro de la iteración. Se concluye que para recomendar procesos de integración que satisfagan las necesidades y tomen ventajas de las fortalezas tanto de Programación eXtrema como de las pruebas de usabilidad remota es necesario más investigación, experimentación y documentación. Finalmente, es necesario afrontar retos que involucran la diferencia en la visión holística de ambos enfoques, así como los problemas de comunicación entre los desarrolladores y los evaluadores de usabilidad, la adaptación de los artefactos para el extremo y la escasa prioridad que se le da a temas de usabilidad.

**Palabras claves:** Programación extrema; pruebas remotas de usabilidad; integración



## Abstract

This study is intended to answer the question of how to integrate remote usability tests into eXtreme Programming. The study is based on a literature review involving search, selection and results classification. Keywords, emerging categories, and search strings in relevant databases were used. An analysis of the current state of the components and processes that have been used in this type of integration has led to a combined approach in areas such as: 1. Requirements management, 2. Management of team members including developers, usability evaluators, experts and final users, 3. Management of tests, data collected and feedback, 4. Design, and 5. Activities carried out within an iteration. It is concluded that more research, experimentation and documentation are necessary to be able to recommend integration processes that satisfy demands and take advantage of eXtreme Programming and remote usability tests. Finally, it is necessary to confront the challenges posed by the difference in the holistic visions of both approaches, as well as communication problems between usability developers and evaluators, adaptation of eXtreme components, and the limited priority assigned to usability efforts.

**Keywords:** eXtreme Programming; Remote Usability Test; Integration; RUT; XP

## Resumo

O objetivo deste trabalho foi responder à pergunta de pesquisa: Como os testes de usabilidade remotos são integrados à Programação Extrema? A metodologia utilizada foi a revisão de literatura, através do processo de busca, seleção e estruturação. Foram utilizadas palavras-chaves, categorias emergentes e redes de pesquisa por meio de base de dados reconhecidos. Como resultado, uma análise do estado atual dos artefatos e processos que foram usados nessa integração, levando a um enfoque combinado em áreas como: 1. Gerenciamento de requisitos, 2. Gerenciamento de membros da equipe, incluindo desenvolvedores, avaliadores de usabilidade, usuários especializados e usuários finais, 3. Manuseio de testes, dados coletados e feedback, 4. Design e 5. As atividades desenvolvidas dentro da iteração. Conclui-se que, para recomendar processos de integração que satisfaçam as necessidades e aproveitem os pontos fortes da Programação Extrema e dos testes de usabilidade remotos, são necessárias mais pesquisas, experimentação e documentação. Finalmente, devemos enfrentar os desafios que envolvem a diferença na visão holística de ambos os enfoques e problemas de comunicação entre desenvolvedores e avaliadores de usabilidade, a adaptação de artefatos para o extremo e escassa prioridade dada aos temas de usabilidade.

**Palavras-chave:** Programação Extrema; Testes de Usabilidade Remotos; Integração

## Introducción

En las últimas décadas las metodologías ágiles han tenido un gran auge (Highsmith, 2002). En el año 1999 Kent Beck publica el primer libro que aborda y define Programación Extrema conocido en el idioma Inglés como eXtreme Programming y referido en adelante con la abreviatura de XP, además menciona los principios ágiles para el desarrollo del *software*, estos

principios centran el enfoque en desarrollo iterativo e incremental para la especificación, el desarrollo y la entrega del producto enfocándose en el proceso, en la participación activa de los usuarios y su interacción con los desarrolladores (Beck, 1999, 2000). Las pruebas de usabilidad remota conocidas en el idioma inglés como Remote Usability Test y referido en adelante con la abreviatura RUT, es un enfoque que ayuda a simplificar y agilizar los procesos de evaluación de



la usabilidad (Bustamante, 2010) dado que evita un proceso laborioso en la configuración de laboratorios, manejo de participantes y administración presencial.

Las metodologías ágiles como XP son ampliamente discutidas por los desarrolladores de *software*, sin embargo se profundiza muy poco cómo integrar los métodos y prácticas de usabilidad, ya que es difícil adaptar las herramientas y prácticas a procesos ágiles, incluso en la literatura disponible generalmente no abarca roles como diseñadores, investigadores de interfaces de usuario y creadores de pruebas (Budwig, Jeong & Kelkar, 2009).

La eficacia de RUT y su adaptabilidad en aplicaciones es ampliamente aceptada por ofrecer resultados similares a los obtenidos en el laboratorio, sumado a un menor costo y una recopilación automatizada de información (Tullis, Fleischman, McNulty, Cianchette & Bergel, 2002). Un ejemplo es la aceptación de RUT en los desarrollos de código abierto, conocidos en el idioma inglés como Open Source Software (OSS), que surge como una alternativa viable, dado que el enfoque de usabilidad ha sido descuidado en las investigaciones de desarrollo de *software* debido a la falta de recursos y métodos de evaluación (Andreasen, Nielsen, Schröder & Stage, 2006).

## Marco teórico

### Conceptos XP y RUT.

La metodología XP presenta un conjunto de conceptos para simplificar el proceso de diseño, desarrollo, prueba y entrega de *software*, en el cual se implementa la simplificación de requerimientos y su entrega incremental (Beck, 2000). Esta metodología se basa en principios ágiles.

Además de entregas incrementales, XP tiene un enfoque basado en reuniones donde se elabora una lista de tareas que van a ser terminadas en la próxima iteración la cual, por lo general no dura más de 4 semanas e involucra a un propietario de producto y equipo de desarrolladores, conjuntamente se fomenta la comunicación (Carnaghan, 2014). La metodología XP asimismo busca el manejo de requerimientos en alto nivel y la identificación de un usuario experto que brinda detalle del requerimiento en la iteración respectiva y el *test* del mismo, basado en pruebas unitarias de código conocido en el idioma inglés como Unit Test y referidas en adelante como UT, pruebas de aceptación de usuario conocido en el idioma inglés como User Acceptance Test y referidas en adelante como UAT y el desarrollo dirigido pruebas, conocido en Inglés como Test Driven Development y referido en adelante como TDD (Lee, 2006).

En este contexto los desarrolladores muestran un bajo interés por las necesidades de los usuarios finales a falta de comprensión y mala interpretación, lo cual lleva a la incapacidad de conectar con los requerimientos de los clientes (Lizano, 2014). Esto ha llevado a problemas de comunicación en dos grandes áreas como son la necesidad de usabilidad ágil y el diseño basado en escenarios para el extremo (Lee, 2006).

Un claro ejemplo se da en el desarrollo de código abierto (OSS), el cual es diseñado y desarrollado para y por los usuarios avanzados de los productos, siendo criticados por su poco o ausente énfasis en la usabilidad (Andreasen, Nielsen, Schröder & Stage, 2006). Una de las técnicas empleadas para conectar con los usuarios y entender su forma de utilizar la aplicación la obtenemos a través de RUT, ya que permite a los desarrolladores de *software* realizar evaluaciones



de usabilidad con los usuarios de una manera práctica y económica (Lizano, 2014), a la vez se goza de los beneficios de dejarlos, de prueba, operar en su entorno normal sin el estrés de una sala llena de equipos de grabación y observadores (Andreasen, Nielsen, Schröder & Stage, 2006).

Asimismo varios estudios han comparado los resultados obtenidos de las pruebas de usabilidad tradicionales con respecto a los resultados obtenidos con RUT, que producen resultados similares, con grandes ventajas como lo son un menor costo, participación geográficamente distribuida, y contribuciones aproximadamente iguales (Madathil & Greenstein, 2011). Las pruebas de usabilidad remota han tenido mucho auge debido a su efectividad como una forma de descubrir problemas críticos de este tipo, también, el contar con usuarios más diversos permite encontrar una gama más amplia de problemas de usabilidad para tipos específicos de usuarios, en entornos específicos (Tullis, Fleischman, McNulty, Cianchette & Bergel, 2002).

RUT es de particular interés para las metodologías ágiles debido a que las iteraciones son cortas, frecuentes y tienen un diseño inicial mínimo, donde el resultado más importante de una prueba de usabilidad es la retroalimentación sobre el sistema que se está evaluando y donde un proceso tradicional de usabilidad no se adapta adecuadamente ya que es costoso tanto financieramente como en tiempo, debido a que requiere de grandes cantidades de diseño inicial, conlleva a un proceso de reclutamiento de participantes laborioso, además de la configuración de un laboratorio (Khatyat, 2009).

### **Retos o problemas encontrados al integrar las metodologías.**

La integración de pruebas de usabilidad en metodologías ágiles presenta una gran resistencia (Lizano, 2014) y falta de recursos para llevar a cabo tareas de usabilidad y enfatizar la importancia de construir sistemas de comunicación para involucrar a los usuarios expertos y novatos y así proporcionar conocimiento y retroalimentación de usabilidad (Andreasen, Nielsen, Schröder & Stage, 2006).

La metodología XP centra su diseño en el usuario, sin embargo no se define la integración con usabilidad y los principios de interacción humano computadora, conocido en inglés como Human Computer Interaction y referenciado en adelante como HCI, con lo cual se plantean varios retos como lo son la integración de conceptos de evaluación de HCI dentro de las iteraciones, las entradas *ad hoc*, la falta de una visión holística en XP y la necesidad de la misma para ejecutar pruebas de usabilidad, las diferencias entre UAT y los *test* de usabilidad, esto sumado a que en XP se le da empoderamiento al desarrollador y el rol de experto en HCI no se menciona como tal (Wolkertorfer, Tscheligi, Sefelin, Milchrahm, Husain, Lechner & Shahzad, 2008).

Especificar por completo un sistema en el momento del diseño no es factible por lo que la adaptabilidad es necesaria tanto en el sistema como por parte de los usuarios donde puedan ajustar su comportamiento para el proceso de diseño evolutivo y participativo, para esto se plantea la necesidad de ayudar a los usuarios comunes en el encuadre del problema, donde se brinden informes en la evaluación remota y así los usuarios reflejen sus problemas y analicen posibles soluciones, de manera que los



desarrolladores y los usuarios trabajarían en ideas de diseño (Draxler & Stevens, 2006).

Otro reto identificado es la falta de tiempo dentro de la iteración para dedicarlo a actividades de interfaz de usuario, la dificultad para crear módulos, así como la priorización de actividades de HCI dentro del desarrollo ha llevado a la necesidad de buscar la optimización dinámica de los desarrolladores y usuarios para la participación en actividades de HCI (Salah, Paige & Cairns, 2014).

Existe la necesidad de crear diseños de sistemas multidisciplinarios centrándose en el *software* y la ingeniería de usabilidad, que cuenten con el respaldo de prácticas y herramientas tanto de XP como de diseño basado en escenario conocido e inglés como Scenario-Based Design y referenciado en adelante como SBD lo que llevaría a escenarios para el desarrollo extremo y principios de usabilidad ágil (Lee, 2006).

Los intereses de los principios ágiles y HCI no están adecuadamente alineados por lo que existe la necesidad de estudios extensos sobre la integración de HCI dentro de metodologías ágiles que incluyan un enfoque iterativo y se explore actividades y artefactos que permitan la integración de pequeñas evaluaciones de usabilidad, para esto se debe cambiar el enfoque de los desarrolladores e incluir pruebas de usabilidad como RUT dentro de las actividades de la iteración (Lizano, Sandoval & Stage, 2014).

El enfoque iterativo de las metodologías ágiles para el desarrollo de *software* son adversas a la inclusión de aspectos de usabilidad lo que presenta un desafío tanto para los desarrolladores, los expertos de HCI y los investigadores en general (Kurosu, 2013).

### Proceso propuestos por diversos autores.

La falta de soporte y recursos para el desarrollo de *software* obstaculiza el integrar las evaluaciones de usabilidad en las metodologías ágiles, sin embargo alguna evaluación es mejor que ninguna, por ejemplo desarrollar prototipos de baja intensidad que lleven a una menor inversión inicial y añadir principios ágiles (Khayat, Hellmann & Maurer, 2010), incorporar la optimización dinámica de los desarrolladores y participantes buscando dar la prioridad necesaria a las actividades de usabilidad, conducen a una mejora en el producto final (Salah, Paige & Cairns, 2014).

El enfoque de desarrollo centrado en el usuario conocido en inglés como User Centered Development y referenciado en adelante como UCD ha sido combinado con metodologías ágiles, por medio de la integración de artefactos y roles como lo son usuarios, herramienta persona, pruebas unitarias, pruebas de usabilidad y evaluaciones de expertos de usabilidad (Wolkerstorfer, Tscheligi, Sefelin, Milchrahm, Hussain, Lechner & Shahzad, 2008). Así mismo, es importante fomentar la participación distribuida que proporcione infraestructura y canales de comunicación centrada en artefactos propios de cada metodología; además, fomentar valores de democracia en el lugar de trabajo, el conocimiento y la experiencia sustentados en reconocimiento, también el compromiso y aceptación de los involucrados (Yetim, Draxler, Stevens & Wulf, 2012).

Dado que XP se enfoca en la calidad del *software* y se basa en iteraciones y la ingeniería de usabilidad se enfoca en los usuarios finales, en cómo estos usan el *software* y hace diferencia entre el que sirve y el utilizable, permite crear una serie de enfoques que combinan la creación de



grupos de trabajo que no solo incluyen a los desarrolladores sino también a los usuarios finales, lo que facilita una mayor colaboración en las iteraciones, así como incluir UT, UAT, RUT y principios de usabilidad en el análisis de requerimientos (Sohaib & Khan, 2010). A esto se le puede sumar el desarrollo de prototipos de baja y alta fidelidad, entrevistas, evaluaciones heurísticas y accesibilidad (de Lima Salgado, Amaral, Freire & Fortes, 2016).

### Metodología

La revisión de literatura es un estudio que identifica, evalúa e interpreta la literatura disponible y relacionada para proporcionar apoyo a una investigación con objetivos específicos (Bailey, Budgen, Turner, Kitchenham, Brereton & Linkman, 2007). Se ha seleccionado este tipo de investigación con

el propósito de contar con un estudio sobre el estado del conocimiento, específicamente para la pregunta base de esta investigación que es ¿Cómo integrar pruebas remotas de usabilidad en la metodología de desarrollo XP? Para responder a esta interrogación se desarrolló una revisión de literatura fundamentada en un proceso de tres pasos: búsqueda, selección y estructuración. A continuación detallamos cada uno de los pasos.

### Búsqueda

Las palabras clave identificadas en la pregunta de investigación y los conceptos alternativos conocidos en el idioma inglés como: eXtreme Programming, Remote Usability Test, Integration, RUT, XP. En la Tabla 1 se muestran las palabras clave, se muestra en la columna 1 el concepto principal y en la columna 2 conceptos alternativos.

Tabla 1  
*Palabras clave utilizadas y sus variantes*

Concepto principal	Conceptos alternativos
RUT	Remote Usability Test Remote Usability Evaluation Remote Usability Assessment.
XP	Extreme Programming
Integration	Integrate

Nota: Fuente propia

La recolección de datos se realizó en el mes de octubre del 2017, en diferentes bases de datos referenciadas en el motor Google Scholar referido en adelante como GS, el cual hace búsquedas en la mayoría de bases de datos especializadas con contenido científico o académico como artículos, tesis, libros, patentes, documentación de congresos y resúmenes. El motor GS hace referencias de búsquedas en las bases de datos tales como: EBSCO, ProQuest, ScienceDirect, Springer, Scopus, IEEE, WoS entre

otras. Además se utilizó el repositorio Exlibri Primo, de uso oficial de la Universidad Nacional de Costa Rica donde se desarrolló este estudio y el cual es utilizado por miles de instituciones en el nivel mundial.

Además de las palabras claves, se utilizó criterios de búsquedas que incluyen rango de años de 1999 al presente, puesto que es a partir de este año que se publica literatura que define la metodología XP (Beck, 1999).



La Tabla 2 consta de cuatro columnas, en la primera columna se muestra la combinación de criterios de búsqueda, en la segunda columna los resultados obtenidos en GS, en la tercera columna los resultados obtenidos en Primo y la cuarta columna muestra el total de los resultados de GS y Primo

por criterio de búsqueda. Al final de la tabla, una fila muestra el total de GS y PRIMO así como el gran total final. Los criterios de búsqueda que no arrojan resultados en las fuentes de búsqueda son descartados y eliminados de la tabla.

Tabla 2  
*Combinación de palabras clave y resultados de búsqueda en GS y PRIMO*

Criterios de búsqueda	Resultados en GS	Resultados en Primo	Total
“RUT” + integration + “XP”	43(1440*)	62	105
“RUT” + integration + “extreme programming”	42	2	44
“RUT” + integrate + “XP”	43(1440*)	15	58
“RUT” + integrate + “extreme programming”	50	0	50
“remote usability test” + integration + “XP”	12	0	12
“remote usability test” + integration + “extreme programming”	4	0	4
“remote usability test” + integrate + “XP”	12	0	12
“remote usability test” + integrate + “extreme programming”	4	0	4
“Remote Usability Evaluation”+ integration + “XP”	26	0	26
“remote usability evaluation”+ integration + “extreme programming”	16	1	17
“remote usability evaluation”+ integrate + “XP”	26	0	26
“remote usability evaluation”+ integrate + “extreme programming”	16	0	16
“remote usability assessment.”+ integration+ “extreme programming “	1	0	1
“remote usability assessment.” + integrate + “extreme Programming “	1	0	1
Total	296	80	376

Nota: Fuente propia

(\*)Dado los resultados de la primera búsqueda descritos en la Tabla 2, nos encontramos con una muestra de 1440 resultados para los criterios de búsquedas “RUT” + integration + “XP” y similar cantidad para “RUT” + integrate + “XP”; debido al tamaño del resultado en la búsqueda de documentos, se decidió aplicar un filtro para tomar en cuenta solo aquellos que hayan sido citados más de cien veces y descartar los demás.

Para agrupar los resultados de la búsqueda realizada en GS y PRIMO se comparan los títulos de los documentos obtenidos para descartar los repetidos y se genera una única lista que da como resultado de búsqueda un total de 121 documentos.

### Selección

Se prosigue con un proceso de es-cogencia de los documentos. Para esto se

consideraron aquellos estudios empíricos, encuestas, casos de estudio, experimentos, revisiones literarias o cualquier otro documento científico formal que describa o aborde el criterio de selección definido: como integrar RUT en XP.

Se leyó cada uno de los 121 documentos con el objetivo de discernir cuales de ellos son accesibles y tratan temas relevantes de acuerdo con las palabras clave y criterio de selección utilizado para su respectiva



evaluación a profundidad, por lo que se seleccionaron 20 documentos que cumplieran con los criterios anteriores para ser incluidos en esta revisión literaria.

### **Estructuración de la revisión literaria.**

El último paso del proceso es la definición de una estructura de la revisión de literatura donde los veinte documentos fueron revisados y documentados mediante la creación de fichas, estas fichas pasaron por un proceso de clasificación mediante la utilización de diagramas de afinidad debido a su amplia utilización para organizar y analizar datos cualitativos (Kawakita, 1991), como resultado se agrupan en tres categorías las cuales definen los conceptos de XP y RUT, los retos o problemas encontrados al integrar ambas metodologías y los procesos propuestos por diversos autores.

### **Análisis de resultados.**

La metodología de desarrollo XP captura los requerimientos en un alto nivel, se basa en iteraciones, y es en cada iteración donde se profundiza en los requerimientos, además busca desarrollar aplicaciones enfocadas en los usuarios por lo que se fomenta la comunicación y la integración de usuarios expertos al equipo de desarrollo. Sin embargo, la aplicación de evaluaciones de usabilidad es muy escasa, lo que ha llevado a que muchas aplicaciones que funcionan no siempre sean usables, esto ha generado la necesidad de investigar e integrar técnicas para incrementar la usabilidad (Beck, 2000; Carnaghan, 2014; Lee, 2006; Sohaib

& Khan, 2010; Salgado, Amaral, Freire & Fortes, 2016). Una de las técnicas utilizadas es RUT, ya que permite a los usuarios realizar evaluaciones de usabilidad con beneficios y resultados similares a las técnicas tradicionales lo que permite conectar a los desarrolladores de *software*, con los evaluadores de HCI y los usuarios finales; esto, de una manera práctica y económica tanto en tiempo como inversión, que aporta además, usuarios y escenarios más diversos (Lizano, 2014; Andreasen, Nielsen, Schröder & Stage, 2006; Madathil & Greenstein, 2011; Tullis, Fleischman, McNulty, Cianchette & Bergel, 2002).

¿Cómo integrar pruebas remotas de usabilidad en la metodología de desarrollo XP? es un tema que ha sido abordado por varios autores, los cuales han propuesto diferentes procesos conocidos también como enfoques combinados, lo que ha generado conceptos como pruebas de usabilidad ágil, desarrollo ágil y usable, prototipos para el extremo, donde además se propone que varios de los artefactos propios de las pruebas de usabilidad sean adaptados para el desarrollo extremo (Salgado, Amaral, Freire & Fortes, 2016; Sohaib, & Khan, 2010; Salah, Paige & Cairns, 2014; Wolkerstorfer, Tschelligi, Sefel, Milchrahm, Hussain, Lechner & Shahzad, 2008; Yetim, F., Draxler, Stevens, Wulf, 2012; Sohaib & Khan, 2010). Estos enfoques combinados han sido resumidos y documentados en la Tabla 3. En la primera columna se muestra el área de enfoque, en la segunda el enfoque de XP, en la tercera columna se muestra el enfoque de RUT y en la cuarta se muestra el enfoque combinado.





Tabla 3  
*Áreas de enfoque para XP y RUT y la propuesta de enfoque combinado*

Área de enfoque	Enfoque de XP	Enfoque de RUT	Enfoque combinado
Desarrollo	Iterativo e incremental	Ejecución de pruebas remotas y captura de datos automatizada	Prototipos simples en cada iteración, pequeñas aplicaciones, aplicar RUT y capturar la retroalimentación.
Requerimientos	Historias, se profundiza en cada iteración	Visión holística, análisis de requerimientos detallados	Historias y requerimientos para usabilidad ágil, analizar problemas en cada iteración
Desarrolladores	Desarrolladores de <i>software</i>	Evaluadores de HCI	Desarrollar grupos de trabajo integrados por desarrolladores de <i>software</i> , evaluadores de HCI
Usuarios	Usuario experto	Usuarios remotos	y usuarios tanto expertos como finales.
Pruebas	UAT, UT, TDD	Pruebas de usabilidad remota	UAT, UT, TDD, RUT, desarrollar el artefacto Persona para el desarrollo Extremo y escenarios para el extremo.
Datos recolectados	Retroalimentación y pruebas automatizadas de código	Recolección de datos automatizado	Reportes, análisis de retroalimentación y datos recolectados de RUT.
Diseño	No se especifica por completo	Especificado	Diseño y pruebas adaptables, mostrar a los usuarios como enmarcar el problema y dar informes.
Actividades	Reuniones diarias, retroalimentación, Retrospectiva	Evaluaciones, análisis	Integrar actividades de RUT en las iteraciones, adaptar artefactos.

Nota: Fuente propia

Por ejemplo, se muestra como en el área de requerimientos en ambos enfoques son compatibles con revisiones en cada iteración, así mismo en el área de desarrollo y usuarios la definición de grupos de desarrolladores y evaluadores. Para los datos recolectados la utilización de pruebas automatizadas puede ser común en ambos enfoques.

En la Tabla 3 se muestran varias propuestas de procesos para integrar RUT con la metodología XP, sin embargo, aún se sigue investigando y tratando de resolver varios retos con diseños de sistemas multidisciplinares, concentrándose en el desarrollo de *software* y la ingeniería de usabilidad, a esto se suma la necesidad de crear sistemas que respalden a través de prácticas y herramientas

los procesos propuestos para integrar RUT con XP. Y es que a pesar de existir un interés por la usabilidad, no se ha logrado dar la prioridad necesaria, esto sumado al entendimiento limitado, falta de recursos como tiempo, apoyo económico y métodos de evaluación, para soportar la integración de RUT con XP. Por lo que se hace necesario incluir RUT como una actividad dentro de la iteración y modificar los artefactos para que puedan ser utilizados en iteraciones (Lizano, 2014; Andreasen, Nielsen, Schröder & Stage, 2006; Wolkerstorfer, Tscheligi, Sefelin, Milchrahm, Hussain, Lechner & Shahzad, 2008; Draxler & Stevens, 2006; Salah, Paige & Cairns, 2014; Lee, 2006; Lizano, Sandoval, & Stage, 2014; Kurosu, 2013).



## Conclusiones

El campo de RUT es ampliamente estudiado, sin embargo, no es incluido como una etapa en la metodología de desarrollo XP. La falta de procesos y herramientas que soporten cómo integrar RUT dentro de XP inicia esta revisión de literatura que nos sugiere el estado actual de dicho esfuerzo y del cual podemos concluir que se necesita mucha mayor investigación, experimentación y documentación para poder definir y recomendar procesos de integración que satisfagan las necesidades y tome ventaja de las fortalezas tanto de XP como de RUT. Para esto, incluir y priorizar actividades de RUT en las etapas de XP y la comunicación de los expertos de HCI con el equipo de XP es fundamental, para diseñar los escenarios ágiles y usables, además de prototipos para el extremo utilizables en las iteraciones proporcionando así, los artefactos adecuados para dichos procesos.

Dentro de los retos encontrados en este trabajo se mencionan áreas que necesitan mayor investigación entre las cuales destacan, cómo manejar la visión holística de los expertos de HCI en contraste con el manejo de requerimientos que se da en XP, el cual, además, se basa en iteraciones que tienen como objetivo entregas rápidas y parciales del sistema, esto genera otro reto en cuanto al manejo de la retroalimentación para ambos enfoques debido al tiempo y momento en el que se recolecta y analiza.

El estado actual nos sugiere cómo diferentes autores han combinado enfoques y artefactos, por ejemplo los desarrollados en la Tabla 3 que incluyen áreas, que con un manejo adecuado y los artefactos necesarios, pueden ser compatibles, entre ellas el desarrollo, los requerimientos, los desarrolladores y los evaluadores, los usuarios, las

pruebas, los datos recolectados, el diseño y las actividades.

Para futuras investigaciones se podría considerar el estado actual de los artefactos de RUT para el desarrollo ágil y cómo estos han sido modificados en comparación con otras metodologías ágiles como Scrum o Kanban. Otros trabajos podrían orientarse a la utilización de XP y RUT para el desarrollo de código abierto, dado los procesos que ambos aportan en el aspecto de desarrollo descentralizado y voluntario, más la necesidad de mejorar y garantizar la experiencia de usuario final.

## Agradecimiento

Agradecer a la Escuela de Informática de la UNA y en especial a Alberto Segura (e. p. d.) que fue pionero en habilitar el espacio necesario para poder presentar este trabajo.

## Referencias

- Andreasen, M. S., Nielsen, H. V., Schröder, S. O., & Stage, J. (2006). Usability in open source software development: opinions and practice. *Information technology and control*, 35(3), 303-312.
- Bailey, J., Budgen, D., Turner, M., Kitchenham, B., Brereton, P., & Linkman, S. (2007). *Evidence relating to Object-Oriented software design: A survey*. In Empirical Software Engineering and Measurement, 2007. ESEM 2007. First International Symposium on (pp. 482-484). IEEE. doi <https://doi.org/10.1109/ESEM.2007.58>
- Beck, K. (1999). Embracing change with extreme programming. *Computer*, 32(10), 70-77. doi <https://doi.org/10.1109/2.796139>
- Beck, K. (2000). *Extreme programming explained: embrace change*. Addison-Wesley professional.
- Budwig, M., Jeong, S., & Kelkar, K. (2009). *When user experience met agile: a case study*. In CHI'09 Extended Abstracts on Human Factors in Computing Systems



- (pp. 3075-3084). ACM. doi <https://doi.org/10.1145/1520340.1520434>
- Bustamante, J. (2010). La herramienta de tests de usabilidad a distancia Loop11. *El profesional de la información*, 19(4). doi <https://doi.org/10.3145/epi.2010.jul.13>
- Carnaghan, I. (2014). *An expeditionary learning approach to effective curriculum mapping formalizing the process by exploring a user-centered framework* (Doctoral Dissertation). Retrieved from University of Baltimore, ProQuest, UMI Dissertations Publishing. (3626119)
- Chalil Madathil, K. & Greenstein, J. S. (2011). *Synchronous remote usability testing: a new approach facilitated by virtual worlds*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 2225-2234). ACM. doi <https://doi.org/10.1145/1978942.1979267>
- De Lima Salgado, A., Amaral, L. A., Freire, A. P., & Fortes, R. P. M. (2016). *Usability and UX practices in small enterprises: lessons from a survey of the Brazilian context*. In Proceedings of the 34th ACM International Conference on the Design of Communication (p. 18). ACM. doi <https://doi.org/10.1145/2987592.2987616>
- Draxler, S., & Stevens, G. (2006). *Getting Out of a Tailorability Dilemma*. In GI Jahrestagung (1) (pp. 576-579).
- Highsmith, J. A. (2002). *Agile software development ecosystems*. Addison-Wesley Professional, Vol. 13.
- Hosseini-Khayat, A., Hellmann, T. D., & Maurer, F. (2010). *Distributed and Automated Usability Testing of Low-Fidelity Prototypes*. In AGILE Conference, 2010 (pp. 59-66). IEEE. doi <https://doi.org/10.1109/AGILE.2010.11>
- Kawakita, J. (1991). *The original KJ method*. Tokyo: Kawakita Research Institute
- Khayat, A. H. (2009). *Distributed Wizard of Oz Usability Testing for Agile Teams*. (Doctoral dissertation). UNIVERSITY OF CALGARY.
- Kurosu, M. (Ed.). (2013). *Human-Computer Interaction: Human-Centred Design Approaches, Methods, Tools and Environments*. 15th International Conference, HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013, Proceedings (Vol. 8004). Springer. doi <https://doi.org/10.1007/978-3-642-39232-0>
- Lee, J. C. (2006). *Embracing agile development of usable software systems*. In CHI'06 Extended Abstracts on Human Factors in Computing Systems (pp. 1767-1770). ACM. doi <https://doi.org/10.1145/1125451.1125784>
- Lizano, F. (2014). *Integrating Usability Evaluations into the Software Development Process: concepts for, and Experiences from, Remote Usability Testing* (Doctoral dissertation). Institut for Datalogi, Aalborg Universitet.
- Lizano, F., Sandoval, M. M., & Stage, J. (2014). *Integrating usability evaluations into scrum: a case study based on remote synchronous user testing*. In International Conference on Human-Computer Interaction (pp. 500-509). Springer, Cham. doi [https://doi.org/10.1007/978-3-319-07233-3\\_46](https://doi.org/10.1007/978-3-319-07233-3_46)
- Salah, D., Paige, R. F., & Cairns, P. (2014). *A systematic literature review for agile development processes and user centred design integration*. In Proceedings of the 18th international conference on evaluation and assessment in software engineering (p. 5). ACM. doi <https://doi.org/10.1145/2601248.2601276>
- Sohaib, O., & Khan, K. (2010). *Integrating usability engineering and agile software development: a literature review*. In Computer design and applications (ICDDA), 2010 international conference on (Vol. 2, pp. V2-32). IEEE. doi <https://doi.org/10.1109/ICDDA.2010.5540916>
- Tullis, T., Fleischman, S., McNulty, M., Cianchette, C., & Bergel, M. (2002). *An empirical comparison of lab and remote usability testing of web sites*. In Usability Professionals Association Conference.
- Wolkerstorfer, P., Tscheligi, M., Sefelin, R., Milchrahm, H., Hussain, Z., Lechner, M., & Shahzad, S. (2008). *Probing an agile usability process*. In CHI'08 Extended Abstracts on Human Factors in Computing Systems (pp. 2151-2158). ACM. doi <https://doi.org/10.1145/1358628.1358648>
- Yetim, F., Draxler, S., Stevens, G., & Wulf, V. (2012). Fostering continuous user participation by embedding a communication support tool in user interfaces. *AIS Transactions on Human-Computer Interaction*, 4(2), 153-168. doi <https://doi.org/10.17705/1thci.00044>



Integración de Pruebas Remotas de sabilidad en Programación Extrema: revisión de literatura (Diego Sánchez-Hernández, Fulvio Lizano-Madriz y María Marta Sandoval-Carvajal) por [Revista Uniciencia](#) se encuentra bajo una [Licencia Creative Commons Atribución-NoComercial-SinDerivadas 3.0 Unported](#).