

Original Research / Artículo original - Tipo 1

A review of the impact on XP methodology of business model inclusion in requirements elicitation

Andrés Felipe Escobar Villada, MSc. (c) / anfelesvillada@gmail.com

Diana Lorena Velandia Vanegas, MSc. (c) / dilove0122@gmail.com

Hugo Armando Ordoñez E., Ph.D. / haordoez@usbcali.edu.co

Universidad San Buenaventura, Cali, Colombia

Carlos Cobos, Ph.D. / ccobos@unicauca.edu.co

Universidad del Cauca, Popayán-Colombia

ABSTRACT XP (eXtreme Programming) agile methodology uses the User Stories as a requirements elicitation strategy. The user stories are based on natural language, which may cause some misunderstanding and miscomprehension problems between software development team and stakeholders. The paper proposes the inclusion of Business Process Models (BPN) in the XP methodology as an alternative to user stories, seeking to improve the quality and quantity of the information collected. The proposal was evaluated using user stories vs. BPN in 11 projects during all phases proposed by XP. Both strategies –and their effectiveness– were analyzed through software metrics, in order to demonstrate the improvement in the development process. By applying software metrics, it was shown that the use of BPN: improves communication between analysts and others involved, increase approval rate of customer requirements, shortens delivery requirement, fewer changes are made in each iteration and a lower percentage of defects are found by the stakeholder, regarding the use of user stories.

KEYWORDS Software; requirements elicitation; eXtreme programming; business process management notation; user stories; software metrics.

Análisis del impacto sobre la metodología XP de la inclusión de modelos de procesos de negocio en la elicitación de requisitos

RESUMEN La metodología ágil de desarrollo de software eXtreme Programming (XP) utiliza Historias de Usuario [HU] como una estrategia de elicitación de requerimientos. Las HU utilizan lenguaje natural, el cual aunque debería ser claro para todos, pues causar confusiones y diferencias en la comprensión de los problemas, entre el equipo de desarrolladores y los interesados [stakeholders]. Este artículo propone incluir el modelamiento de procesos de negocio [PN] en la metodología XP, como una alternativa a las historias de usuario capaz de mejorar la calidad u cantidad de información recolectada. En el desarrollo de esta investigación se evaluó un total de once proyectos (HU vs PN), durante todas las fases propuestas por XP. Las dos estrategias y su efectividad fueron analizadas a través de métricas de software para demostrar el mejoramiento en el desarrollo de los procesos. El trabajo realizado evidenció que el uso de PN mejora la comunicación e incrementa la calidad del proceso y la productividad del equipo de desarrollo.

PALABRAS CLAVE Software; elicitation de requerimientos de software; XP; BPMN; historias de usuario; métricas de software.

Análise do impacto sobre a metodologia XP da inclusão de modelos de processos de negócio na elicitação de requisitos

RESUMO A metodologia ágil de desenvolvimento de software Extreme Programming (XP) utiliza Histórias de Usuário [HU] como uma estratégia elicitação de requisitos. As HU usam linguagem natural, que embora devesse ficar claro para todos, pode causar confusão e diferenças na compreensão de problemas entre a equipe de desenvolvimento e as partes interessadas [skateholders]. Este artigo propõe a inclusão da modelagem de processos de negócios [PN] na metodologia XP, como uma alternativa às Histórias de Usuário, capaz de melhorar a qualidade ou quantidade de dados coletados. No desenvolvimento deste trabalho foram avaliados um total de onze projetos (HU vs PN) em todas as fases propostas pelo XP. As duas estratégias e a sua eficácia foram analisadas através de métricas de software para demonstrar o melhoramento no desenvolvimento dos processos. O trabalho mostrou que o uso de PN melhora a comunicação e aumenta a qualidade do processo e a produtividade da equipe de desenvolvimento.

PALAVRAS-CHAVE Software; elicitação de requisitos de software; XP; BPMN; histórias de usuários; métricas de software.

I. Introduction

Client requirements and mandatories management requires every company to have a base on software tools, which can be acquired on demand. Thus, software development worldwide is becoming a highly important asset since it allows company evolution and knowledge management (Christensen, 2014; Paternoster & Giardino, 2014). However, the terms and conditions for every project are different, which means the methodology must necessarily be adapted to each project or environment. Hence, agile methodologies are specially conceived for projects where software development includes conditions such as critical deadlines and volatile or new technologies based requirements (Pino, 2010; Losada & Urretavizcaya, 2013). One of those technologies is *eXtreme Programming (XP)*, which strongly recommends team working and a straightforward feedback between the parties involved: clients (interested project users) and the development team (Thiyagarajan & Verma, 2009). In the process of an agile methodology-based project software (e.g., XP), information should be obtained in a fast, clear and comprehensive way for all the people involved in the development process. It is critically important that the client consultant provides the development team with as much information as possible in short periods of time (Schneider, Naughton, & Berenbach, 2012). However, many details are left out of this task, which does not make for a well-defined project approach, and requires many changes to be made continuously along the way. Overall, software development companies are greatly affected by this issue since they need to pursue a strict budget for each project (Joskowicz, 2008).

XP includes a direct communication between involved parties and is built in order to work in software projects developed by small teams of programmers (Beck, 2000). In XP, requirements elicitation is performed through the use of user stories, where cards describe client needs. Nonetheless, this description is executed by hand in a natural language, which is open to mistakes, since the information tends to be ambiguous and vague (Al-Zoabi, 2008; Domann, 2014). Additionally, user stories are modified and refined each time an iteration is performed with the target on even better estimates (Suaza, 2013). Hence, the requirements might be misunderstood and may not cover the needs of all the involved parties (Chinosi & Trombetta, 2012). Often, conversations at meetings between the development team and involved par-

I. Introducción

La gestión de las exigencias y requisitos de los clientes requiere que las organizaciones se soporten en herramientas software, las cuales pueden ser adquiridas o desarrolladas a medida. En ese sentido, el desarrollo de software se constituye en parte importante del capital mundial, debido a que permite la evolución de las empresas y la administración del conocimiento (Christensen, 2014; Paternoster & Giardino, 2014). Sin embargo, las condiciones y limitaciones de cada proyecto son diferentes, lo cual hace necesario adaptar la metodología a cada proyecto o situación. En este sentido, las metodologías ágiles han sido especialmente pensadas para proyectos donde el desarrollo de software tiene condiciones como: plazos reducidos y requisitos volátiles o basados en nuevas tecnologías (Pino, 2010; Losada & Urretavizcaya, 2013). Una de estas metodologías es *eXtreme Programming (XP)*, la cual promueve el trabajo en equipo y propicia una retroalimentación continua entre los involucrados: clientes (o usuarios interesados en el proyecto) y el equipo de desarrollo (Thiyagarajan & Verma, 2009). En el desarrollo de un proyecto de software basado en una metodología ágil (e.g., XP), la información debe capturarse de manera rápida, clara y comprensiva para todos los involucrados en el proceso de desarrollo. Es vital que el grupo de desarrollo pueda captar la mayor cantidad de información ofrecida por el representante de los clientes como sea posible, en periodos cortos de tiempo (Schneider, Naughton, & Berenbach, 2012). Sin embargo, en esta tarea se dejan muchos detalles por fuera, lo que impide tener un alcance del proyecto bien definido, y hace que el proceso esté abierto a cambios continuos. Tema que afecta a las empresas de desarrollo de software, las cuales deben ceñirse a un presupuesto específico por proyecto (Joskowicz, 2008).

XP maneja una comunicación directa con los involucrados y está diseñada para trabajar con proyectos de software que serán desarrollados por equipos pequeños de programadores (Beck, 2000). En XP, la elicitación de requisitos se realiza a través del uso de historias de usuario, donde por medio de tarjetas se describen las necesidades del cliente. Sin embargo esta descripción se realiza en papel y en lenguaje natural, el cual por sí mismo es propenso a errores, dada su ambigüedad y la vaguedad de la información recolectada (Al-Zoabi, 2008; Domann, 2014). Además las historias de usuario son modificadas y refinadas cada vez que se realiza una iteración, con el propósito de lograr una mejor estimación (Suaza, 2013). Esto puede causar que los requisitos no sean identificados de manera clara y no cubran todas las necesidades de los involucrados (Chinosi & Trombetta, 2012). A menudo, en las reuniones entre los desarrolladores y los involucrados, el diálogo para construir las historias de usuario no es el adecuado, lo que genera que el producto de software, al final, no satisfaga sus expectativas (Zheng, 2012).

En consecuencia, existe la posibilidad de desviarse del objetivo de la historia de usuario, la cual es centrarse en la tarea; la historia de usuario solamente debe describir comportamientos externos del sistema que puedan ser entendidos por el cliente (Tong, 2010). El uso de un lenguaje de representación visual de

procesos de negocio [PN], modelados con *Business Process Model and Notation* [BPMN], facilita la documentación del proceso, a través de un modelo visualmente entendible para todos los actores que intervienen en un proyecto de construcción de software (Wong & Gibbons, 2011).

Por lo anterior, en este artículo se propone incluir PN, como una estrategia que permita mejorar la comunicación entre los involucrados en un proyecto de desarrollo de software, en la metodología XP. Al mencionar que se propone una estrategia para la elicitación de requisitos con modelos PN, se hace referencia a proponer una serie de pasos o elementos que tienen como finalidad la consecución de un determinado objetivo, en este caso, el uso de modelos PN en lugar de las historias de usuario. La estrategia será validada a través de una serie de métricas que ayudan a identificar los elementos en los que aporta y logra la mejora sobre la metodología los modelos PN. En la sección dos se presentan algunos de los trabajos más representativos en la temática de investigación; en la sección tres, la estrategia propuesta; en la sección cuatro, las hipótesis planteadas para la validación; en la sección cinco, los resultados obtenidos en la experimentación con los proyectos en desarrollo; y, finalmente, en la sección seis, las conclusiones y trabajo futuro.

II. Trabajos relacionados

Según Blom (2010), las metodologías ágiles están centradas en potenciar las relaciones interpersonales como punto clave para el éxito en el desarrollo de software. En este sentido, la comunicación constante con los involucrados es vital para crear la historia de usuario, describiendo claramente las necesidades y los objetivos de la organización (Newkirk, 2002). Aunque existen varias plantillas sugeridas respecto de la información que debe contener una historia de usuario, no hay un consenso al respecto. Jaqueira, Lucena, Aranha, Alencar, y Castro (2013) afirman que la mayor debilidad de las historias de usuario es que los administradores, usuarios y desarrolladores requieren creer en la práctica de cosas simples, plasmadas y descritas en lenguaje natural, sin tener en cuenta que esto genera volatilidad de la información recolectada. La historia de usuario solamente debe describir comportamientos externos del sistema, que puedan ser entendidos por los involucrados.

Zheng (2012) pretende examinar qué tanto puede adaptarse una metodología ágil en la implementación de un BPMS en un proyecto complejo, analizando la implementación total del proyecto y su impacto en la organización. Su trabajo se enfocó particularmente en la metodología ágil *Scrum*; en la cual el equipo se conforma por el personal de desarrollo y los involucrados, y al final de cada iteración, los involucrados evalúan los entregables (y su evaluación se documenta); con esto el analista de negocio comienza a diseñar un diagrama de procesos a través de la notación BPMN. Por otra parte Ottensooser, Fekete, Reijers, Mendling, y Menictas (2012), plantean un experimento que compara una plantilla de casos de uso frente a la notación gráfica de procesos de negocio BPMN que representan los requisitos del sistema, con el fin de entender y

ties are not effective, which means that the final software product does not fulfill all expectations (Zheng, 2012).

In consequence, there is a risk of going off topic on the user story, which should focus only on the task; the user story should describe only the system's external behaviors that the client is able to understand (Tong, 2010). The use of a business process visual representation language [BPN], obtained by *Business Process Model and Notation* [BPMN], helps the process documentation through a clear visual model that all involved parties are able to understand in a software construction project (Wong & Gibbons, 2011).

Therefore, this article aims to include BPN as a strategy to strengthen the communication between involved parties on a software development process using XP methodology. When a strategy is proposed for requirements elicitation using BPN models, there is reference to a set of stages or elements with the aim of achieving an objective, for instance, using BPN models instead of user stories. The strategy will be validated through a set of metrics, which help to identify the elements where BPN models contribute and achieve improvements to the methodology. Section II shows several representative approaches to the subject under present investigation; Section III gives the proposed strategy; Section IV exhibits the validation hypothesis; Section V expresses the results obtained from the development project experiments; finally, Section VI presents conclusions and future work.

II. Related work

According to Blom (2010), agile methodologies are focused on strengthening interpersonal relationships as a key point for success in software development. Thus, constant communication between involved parties is crucial in order to create user stories by drawing clearly the needs and targets of the organization (Newkirk, 2002). Although there are several proposed templates in relation to the information allocated as a user story, there is no point of agreement. Jaqueira, Lucena, Aranha, Alencar and Castro (2013) state that the major weakness in user stories involves managers, users and developers remaining attached to easy practice, which is described in natural language and makes the collected data volatile. Each user story should only trace external system behavior that might be comprehensible for all involved parties.

Zheng (2012) aims to evaluate how much an agile methodology could be adapted for a BPMN implementation in a complex project by studying the whole project implementation and its impact on the organization. That research is focused on the agile methodology *Scrum*, where the team is composed of the development team and the involved individuals. At the end of each iteration, all the individuals review the final results (the test documentation is executed). Following this process, the business consultant starts to draw a process diagram using BPMN notation. On the other hand, Ottensooser, Fekete, Reijers, Mendling, and Menictas (2012) conduct an experiment by comparing a template of use cases against BPMN business process graphic notation, which represents the system requirements, in order to better understand and represent the processes in a development project. The experiment concludes that graphic notation allows an even faster cognitive understanding and the use cases help many process exceptions to come to light. Furthermore, the understanding of the involved parties and analysts is improved when a use cases template is first put on the table, that is, first a textual model and later its equivalent in BPMN, a graphic model.

Ordoñez et al. (2015) study the impact of BPN model utilization instead of user stories on the requirements stage of the XP methodology. The results achieved show that BPN models help to improve both the quality and amount of data collected during the requirements elicitation stage of eleven software projects, though other XP methodology stages were not considered.

Taking the analysis of the mentioned studies into account, it is clear that the use of natural language between involved parties and the development group frequently means that the approaches and priorities to the requirements are not described well, which generates delays in development, following the redefinition and reviews of each user story. Furthermore, when gathering information, the involved individuals tend to skip data which is obvious for them, allowing conflicts to arise between the requirements and the needs of the involved parties. This article is intended to evaluate the inclusion of BPN models in an agile development methodology such as XP instead of user stories, including metrics analysis such as the number of approved requirements at each iteration, deadlines, changes in requirements and issues or bugs found by the client. BPN implementation allows all the involved parties to use an easy and standard notation,

representar de mejor manera los procesos de un proyecto a desarrollar. El experimento concluyó que la notación gráfica permite un entendimiento cognitivo más rápido y que los casos de uso permiten ver muchas excepciones en el proceso. Por otra parte, el entendimiento para los involucrados y analistas mejora cuando se aborda primero la plantilla de casos de uso, es decir un modelo textual, y luego su equivalente en BPMN, un modelo gráfico.

Ordoñez et al. (2015) analizan el impacto de usar modelos de PN en lugar de historias de usuario en la fase de requisitos de la metodología XP. Esta propuesta logró evidenciar que los modelos de PN permitieron mejorar la calidad y cantidad de información recolectada durante la fase de elicitación de requisitos en once proyectos de software, pero no se abordaron las fases siguientes de la metodología XP.

Con base en el análisis de los trabajos citados, se identifica que la utilización de lenguaje natural entre involucrados y grupo de desarrollo hace que, en ocasiones, no se definan bien los alcances o las prioridades de los requisitos, lo que genera retrasos en el desarrollo, por la revisión y redefinición de los requisitos en cada historia. Además, los involucrados, al capturar la información a través de las historias de usuario, omiten información que consideran “obvia”, lo que genera requisitos que están en conflicto con las necesidades de los involucrados. En este artículo, se propone evaluar la inclusión de modelos de PN en una metodología de desarrollo ágil como XP en lugar de historias de usuario, a través del análisis de métricas como el número de requisitos aprobados en cada iteración, el tiempo de entrega, los cambios incorporados a requisitos y los defectos encontrados por el cliente. El uso de PN permite a los involucrados adoptar una notación simple y estándar, que facilita la captura de requisitos con un modelo visual y claramente entendible, respaldando las conversaciones entre el equipo de desarrollo y los involucrados, con el propósito de aumentar la productividad del equipo de desarrollo y la calidad de los proyectos software a desarrollar. La calidad en la descripción de los requisitos (Dorigan & Miranda-de-Barros, 2014) en este trabajo se logra al generar modelos de PN construidos con los involucrados dentro del contexto del requisito y con la descripción de todas las tareas, actividades, flujos y roles que intervienen en el proceso de desarrollo.

III. Estrategia propuesta

La propuesta busca realizar una representación visual de los requisitos generados en la metodología XP, a través de modelos de PN definidos con la notación BPMN, que reemplace las historias de usuario. Para la definición de la propuesta, se toma cada elemento de la historia de usuario y se establece la correspondencia en la notación BPMN. En este trabajo los PN son representados usando BPMN 2.0 (Object Management Group, 2006). La **FIGURA 1** hace una representación gráfica de la relación entre cada uno de los elementos de una historia de usuario y los elementos dentro de un modelo de PN.

A. El manifiesto ágil y el uso de modelos PN

XP es una de las llamadas metodologías ágiles de desarro-

llo de software más exitosas en tiempos recientes (Joskowicz, 2008). XP está diseñada para entregar el software a los clientes en el momento en que lo necesitan. Al ser una metodología ágil debe cumplir con el manifiesto ágil el cual establece cuatro principios: se valora a los individuos y las interacciones sobre los procesos y las herramientas; se valora a las aplicaciones que funcionan sobre la documentación exhaustiva; se valora la colaboración del cliente sobre las negociaciones contractuales; y se valora la respuesta al cambio sobre el seguimiento de un plan (Beck et al., 2001).

En este sentido, la inclusión de modelos de PN como estrategia que reemplace las historias de usuario aporta más formalidad al proceso, y puede generar la violación de los principios de la metodología; sin embargo, pensar únicamente en el diseño de lo que se debe entregar inmediatamente, sin tener en cuenta lo que deberá realizarse inmediatamente después o dentro de poco tiempo, no siempre es la mejor decisión. Es cierto que muchas veces, por intentar generalizar o prever futuros casos, se invierte tiempo extra para entregables que quizás no se requerirían, sin embargo, si el análisis fue correctamente realizado, este tiempo extra inicial redundará en un menor tiempo total (Joskowicz, 2008).

Los modelos PN realizan la codificación de políticas, normas, leyes y mejores prácticas que son utilizadas por una organización para tomar decisiones (Fair Isaac Corporation, 2010). En XP las historias de usuario sustituyen a los documentos de especificación funcional y a los “casos de uso”. Estas “historias” son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar (Joskowicz, 2008). Sin embargo, la participación del cliente se orienta a la constante retroalimentación que propone XP. En esta propuesta, la retroalimentación se registrará de manera gráfica a través de la documentación del proceso en un PN, con el fin de realizar estimaciones de tiempo y esfuerzo de manera más sencilla, y facilitar al analista la revisión de los requisitos o cambios generados. Stephens (2002) aclara que la baja documentación puede afectar el posterior mantenimiento del sistema; si bien durante el proyecto el equipo tiene en mente todas sus particularidades, hay que prever que pasará luego de ser entregado, cuando el equipo se disuelva y sea necesario realizar algún cambio o mejora. La notación de los requisitos a través de modelos PN permite unificar y definir los requisitos y la trazabilidad de los cambios a lo largo del proyecto.

Finalmente, los modelos PN, aunque agregan un componente formal a la documentación de los requisitos, se proponen como estrategia por ser una notación fácil de entender, no solo para los analistas, sino para los involucrados. Y aunque puede generar mayor tiempo en la primera iteración de la metodología, este se verá reducido en fases como la de planificación o ejecución, en donde no se tiene que validar con el cliente y realizar recodificación.

B. Correspondencia del PN en la historia de usuario

El ciclo de vida de un proyecto XP es dinámico. Por esto, se trata de realizar ciclos de desarrollo cortos (llamados iteraciones), con entregables funcionales al finalizar cada ciclo

which helps to address the requirements with a visual model that is clearly comprehensible, and by supporting the development team’s conversations with involved individuals, in order to enhance the team’s productivity and the quality of the software projects. The quality of requirements description is achieved through the production of BPN models built with the help of the involved parties in the requirements context, and the fine description of tasks, sets of activities, flows and roles within the development process.

III. Solution strategy

The solution looks toward performing a visual representation of requirements generated using the XP methodology across BPN models described with BPMN notation, which replaces user stories. For solution definition, each element of the user story is mapped with its related match in BPMN notation. In this study the BPN models are represented using BPMN 2.0 (Object Management Group, 2006). **FIGURE 1** plots the relation between each element in a user story by contrast with the elements of a BPN model.

A. The Agile Manifesto and use of BPN models

Nowadays, XP is one of the most successful agile methodologies for software development (Joskowicz, 2008). It is designed to deliver software at the right moment when clients need it. Given that it is an agile methodology, it must follow the Agile Manifesto, which is built on four principles: evaluation of individuals and interactions over processes and tools; evaluation of working and functional applications over exhaustive documentation; evaluation of client contribution on contract deals; and the evaluation of the response to changes over a schedule or plan follow-up (Beck et al., 2001).

Therefore, the inclusion of BPN models as a strategy to replace user stories contributes greater reliability in the process, which may have a major impact on the methodology principles. Nonetheless, it is not the best approach to think only about the design of the immediate results without bearing in mind the tasks to perform in a short time. On many occasions, extra time is needed for unrequired deliverables in order to take a wider approach or to prevent future changes, though if the formulation is correctly done, this extra time could end up making the total time shorter than expected (Joskowicz, 2008).

The BPN models provide definition of the policies, guidelines, laws and best practices required by any organization in order to make decisions (Fair Isaac Corporation, 2010). In XP, user stories are a substitute for functional specification documents and “use cases”. The client writes these “stories” in his own language as a short description of what the system should perform (Joskowicz, 2008). However, the client contribution is addressed to the continuous feedback that XP suggests. With this recommendation, feedback is recorded in a graphic way throughout the process documentation in a BPN, with the purpose of making time and effort estimates in a simple way that helps the analyst to check for requirements and sudden changes. Stephens (2002) suggests that briefer documentation may further impact system maintenance; given that the development team must bear in mind all details across the whole project, what might happen after delivery must also be considered, even when the team is disbanded but it is then necessary to make some change or improvement. Requirements notation through BPN models allows requirements to be defined and merged and changes to be tracked within the project.

Finally, though BPN models add a formal component to requirements documentation, they are planned as a strategy since they are in easily comprehensible notation, for not only the analyst but also the involved individuals. Although more time might be required at the very first iteration of the methodology, it will be split into stages such as planning or execution, where it does not stop for client validation or rewriting.

B. BPN correspondence with user story

An XP project life cycle is dynamic. In consequence, it is intended to perform short development cycles (called iterations), which include functional deliverables at the end of each cycle (Joskowicz, 2008). Each iteration executes a full cycle of analysis, design, development and tests, using a set of rules and practices that are described in XP. Taking this into account, Pohl’s (2010) statement is highlighted, that requirements elicitation is performed as a transversal process to development, as well as XP states on iterations.

XP includes four stages: exploration, planning, iterations and production. The present study is focused only on the exploration stage since the target is to enhance productivity and software quality from the starting point of requirements elicitation. In this exploration stage, the

(Joskowicz, 2008). En cada iteración se realiza un ciclo completo de análisis, diseño, desarrollo y pruebas, pero utilizando un conjunto de reglas y prácticas que caracterizan a XP. En este sentido, se toma como referencia lo planteado por Pohl (2010), donde la elicitación de requisitos se realiza como un proceso transversal al desarrollo, tal como lo plantea XP en sus iteraciones.

En XP se abarcan cuatro fases de desarrollo: exploración, planificación, iteraciones y puesta en producción. En este trabajo se aborda solamente la fase de exploración, ya que el objetivo es aumentar la productividad y calidad del software, partiendo de la elicitación de requisitos. En la fase de exploración los involucrados definen lo que necesitan mediante la redacción de “historias de usuario” a alto nivel y los programadores estiman los tiempos de desarrollo con base en esta información (Beck, 2000).

Para la definición de la propuesta se toma cada elemento de la historia de usuario y se establece la correspondencia en la notación BPMN. La **TABLA 1** y la **FIGURA 1** detallan cómo se representa cada elemento de la historia de usuario en un PN.

IV. Métricas

Para valorar el éxito de un proyecto de desarrollo de software es necesario medir, solo así se puede analizar su estado real, analizar los riesgos potenciales en cada momento, ajustar el flujo de trabajo y controlar la calidad del producto desarrollado, entre otras cosas. En un estudio desarrollado por Ordoñez et al., (2015) se aplicaron métricas solo al proceso de elicitación de requisitos para verificar el aporte de los modelos PN en la metodología. En este artículo se aplican diversas métricas de software a la metodología XP (en su totalidad), para analizar el impacto del uso de los modelos PN en lugar de la historia de usuario.

Con base en el estudio realizado para construir el estado del arte, en el proceso de esta investigación se usaron las métricas descritas en la **Tabla 2**, para validar la estrategia propuesta. Con esto se buscó identificar el aporte de los PN en la metodología XP, en lugar de las historias de usuario en todo el proceso de desarrollo. La mayoría de las métricas derivan de las herramientas propias de *Scrum* (i.e., lista de requisitos priorizada, lista de tareas de la iteración y gráficos de trabajo pendiente) en donde, para agilizar el análisis, se utiliza el *agile balanced scorecard* [cuadro de mando ágil] (**FIGURA 2**), en el que se combinan varias métricas diferentes pero relacionadas entre sí (Kazi, Radosav, Nikolic, & Chotaliya, 2011; (Mukker, Singh, & Mishra, 2014).

La selección de las métricas se basó en aquellas que permitan tomar decisiones sobre los resultados del proyecto y la satisfacción del cliente a partir de requisitos implementados y aprobados. Por lo tanto, fue conveniente identificar en el cuadro de mando los objetivos del proyecto, desde la fase de elicitación de requisitos (exploración), hasta la planificación y las iteraciones que generaba cada entrega. Es importante mostrar entonces, por ejemplo, la velocidad con que se consigue implementar y aprobar los requisitos del proyecto, y guiar e informar sobre

Table 1. User story elements match in BPMN notation / *Tabla 1. Correspondencia de los elementos de las historias de usuario en notación BPMN*

Item	User Story / <i>Historia de usuario</i>	BPMN
Title / <i>Título</i>	States the main objective / <i>Plantea el objetivo principal</i>	BPN model name. / <i>Nombre del modelo de proceso de negocio.</i>
Description / <i>Descripción</i>	Description of activities supported by involved parties. / <i>Descripción de las actividades que aportan los involucrados.</i>	Defined by the name and attached description of each of the tasks that completes the logic sequence of the BPN model. / <i>Definida por el nombre y la descripción adjunta a cada una de las tareas que conforman la secuencia lógica del modelo PN definido.</i>
History review / <i>Revisión histórica</i>	Allows tracking of changes in requirements by describing the one who is responsible, and changes the date. / <i>Permite realizar trazabilidad a los cambios que se realicen a los requisitos, estableciendo el responsable y fecha de los cambios.</i>	Changes and their tracking are represented through process versions, which allow attempting to circumstances such as: rules changes, notifications and exceptions. / <i>Los cambios y la trazabilidad de los mismos se representarán a través del versionamiento del proceso, el cual permita responder a circunstancias tales como: cambio en las reglas, notificaciones y excepciones.</i>
Reference documents / <i>Documentos de referencia</i>	Elements that allow new requirements to be elicited or inbound/outbound products to be made a requirement. / <i>Elementos que permiten elicitar nuevos requisitos o productos de entrada o salida dentro de un requisito.</i>	Data objects, where the In/Out process document is written. / <i>Objetos de datos, donde se especifica el documento de entrada o salida de un proceso.</i>
Involved actors / <i>Actores</i>	Describes who is involved in the current documented requirement. / <i>Describe quiénes intervienen en el requisito que se está documentando.</i>	They are represented by roles in each project. The Lanes help to clearly establish each role's responsibilities for each activity or process tasks. The Pools identify the process. / <i>Se representan a través de los roles dentro de cada proceso. Los Carriles (Lanes) permiten establecer claramente las responsabilidades de cada rol, para cada actividad o tarea del proceso. Las Piscinas (Pools) permiten identificar el proceso.</i>
Dependencies / <i>Dependencias</i>	Indicate prior user stories for the described activities, with the aim of establishing prior processes or requirements that should be executed. / <i>Se indican las historias de usuario previas a las actividades que en ella se plantean, con el fin de establecer requisitos o procesos previos que deban ejecutarse.</i>	They are determined by forwarding messages between processes and the creation of sub-processes in order to clarify the sequence of activities among different roles and tasks. / <i>Se determinan con el envío de mensajes entre procesos y la creación de subprocesos que permiten indicar la secuencia de actividades entre diferentes roles y tareas.</i>

HISTORIA DE USUARIO	NOTACIÓN BPMN	SÍMBOLO
Título de Historia	Nombre del Proceso	
Revisión Histórica	Versionamiento del Proceso	
Documentos de Referencia	Objetos de Datos , pero si el documento no hace parte del proceso sino como referencia para elicitar se trata como anotación.	
Actores	Roles. Asignación de Tareas a través de Lanes.	
Dependencias	Subprocesos o Procesos relacionados. Envío de Mensajes entre procesos.	
Descripción de las Tareas	Secuencia de Actividades	

Figura 1. User Story vs BPN Models Match / *Figura 1. Correspondencia entre Historia de Usuario vs Modelos de PN*

involved parties determine what they need through “user stories” written at a high level, and the programmers are then open to make time estimations for development based on this information (Beck, 2000).

For the solution definition, each element is taken from the user story and its match in BPMN notation is established. **TABLE 1** and **FIGURE 1** both describe how each element of a user story is described in a BPN.

IV. Metrics

In order to evaluate the success of a software development project it is necessary to make measurements, as its real state can be assessed only in this way, in order to analyze the potential risks at each moment, to fix the workflow and to control the developed product quality, among others. Ordoñez et al. (2015) conducted a study in which metrics were applied only to the requirements elicitation process, with the focus on chec-

Table 2. Research supporting metrics / Tabla 2. Métricas que soportan la investigación

Metric / Métrica	Type / Tipo	Use / Uso
M1: Delivery time of a requirement after request. / M1: Tiempo de entrega de un requisito tras su petición.	Metrics for productivity and delivery effectiveness / Métricas de productividad y efectividad de la entrega	Delivery time difference (days) with the planned time for iteration with both user stories and BPN models. / Diferencia de tiempo de entrega (en días) con el tiempo de planificación por iteración con historias de usuario y modelos PN.
M2: Completed requirements on iteration. / M2: Requisitos completados en la iteración	Metrics for project results / Métricas de resultados del proyecto	Ideally, this should increase over time (productivity). It depends on approved requirements at each iteration. The number of approved requirements at each iteration with both user stories and BPN models will be evaluated. / Idealmente debería aumentar con respecto al tiempo (productividad). Depende de los requisitos aprobados en cada iteración. Se evaluará el número de requisitos aprobados en cada iteración con historias de usuario y con modelos PN.
M3: Injected changes and added requirements over initial project approach. / M3: Cambios incorporados y requisitos añadidos sobre el alcance inicial del proyecto.		Number of changes created to the requirements once an iteration is finished for those elicited with both user stories and BPN models. / Número de cambios generados sobre los requisitos finalizada una iteración de aquellos elicitados con historias de usuario y modelos PN.
M4: Bugs (issues found by client or product users), by state and critics. / M4: Incidencias (defectos encontrados por el cliente o usuarios del producto), por estado y por criticidad	Functional quality / Calidad funcional	Metric that allows an approach to the number of changes to requirements found by the client using both user stories and BPN models. / Métrica que permite medir el número de modificaciones sobre los requisitos encontrados por el cliente con historias de usuario y modelos PN.

king the contribution of the XP models to the methodology. This article shows several software metrics applied to the XP methodology (as a whole) with the aim of studying the impact of the use of BPN models in place of user stories.

Based on the study performed in order to build the state of the art, in the process of this research the metrics described in TABLE 2 were used, which helps to validate the proposed strategy. Hence, it was intended to try to identify the contribution of BPN in XP methodology, instead of user stories, across the whole development process. Most of the metrics are inherited from Scrum tools (i.e., prioritized requirements list, iteration tasks list and pending work plots), where the agile balanced scorecard is used in order to optimize the study (FIGURE 2), where several metrics are merged but related to each other (Kazi, Radosav, Nikolic, & Chotaliya, 2011; Mukker, Singh, & Mishra, 2014).

Metrics selection was based on those that allow decisions to be made on project results and client satisfaction with the sight of the implemented and approved requirements. Hence, this selection is convenient for the project management board to identify metrics, from the requirements elicitation stage (exploration) to planning and iterations for each deliverable. For instance, it is very important to present the pace at which implementation will be performed, as well as the project requirements approval, and it is also important as a guide on which proposed strategy

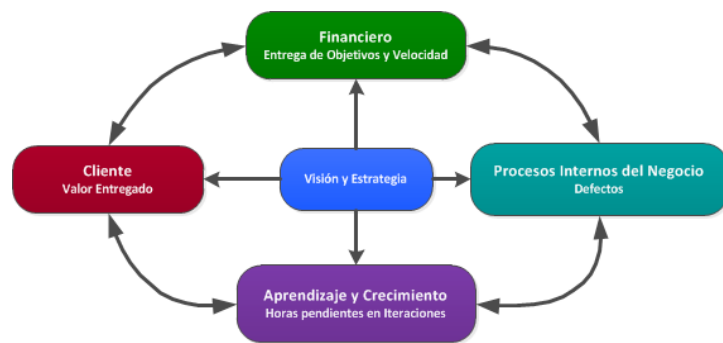


Figure 2. Agile balanced scorecard (Kazi et al., 2011) / Figura 2. Cuadro de mando ágil (Kazi et al., 2011)

cuál de las estrategias propuestas lo está logrando en mayor medida. Las métricas que no fueron tenidas en cuenta en este estudio tienen que ver con la mantenibilidad y la situación financiera, dado que no son relevantes para su objetivo. La TABLA 2 detalla las métricas que fueron aplicadas a los once proyectos que usaron de manera indistinta cada estrategia: historias de usuario o modelos de PN.

El objetivo de las métricas en esta investigación es permitir identificar cuál de las estrategias ayuda a mejorar el entendimiento entre los involucrados y los analistas, a partir de la cuantificación de los requisitos aprobados y del tiempo que toma la aplicación de cada estrategia en las fases de planificación e iteraciones, y no solo de exploración. Los resultados permitieron reflejar que hay un aumento en la calidad del producto final y en la productividad del equipo de desarrollo al elicitar requisitos más comprensibles por los involucrados.

V. Validación y resultados

Para el análisis de las preguntas planteadas –y con el fin de comparar la cantidad y calidad de información recolectada en cada

una de estas-, los analistas usaron grupos focales con los involucrados, después del uso de las historias de usuario o modelos PN. Los grupos focales son una técnica de recolección de datos mediante una entrevista grupal semiestructurada que gira alrededor de una temática propuesta (Escobar & Bonilla-Jimenez, 2009); su propósito principal es lograr que surjan inquietudes y reacciones en los involucrados y analistas. Seguido a la aplicación de los grupos focales –como herramienta de análisis de resultados que permite evaluar los efectos del uso de la estrategia propuesta respecto de las historias de usuario–, se utilizaron los casos de estudio, útiles para investigar un solo fenómeno o entidad dentro de un tiempo determinado. Los casos de estudio ayudan a la industria a evaluar los beneficios de los métodos y las herramientas, y proveen una manera efectiva de asegurar que los cambios en los procesos generan los resultados deseados. Por definición, los casos de estudio se llevan a cabo en entornos del mundo real, por lo que, independientemente del nivel de control aplicado, tienen un alto grado de realismo.

Con los proyectos utilizados para la validación de esta propuesta, se aplicó en paralelo las dos estrategias (historias de usuario vs PN) y se procedió a aplicar las métricas descritas en la **TABLA 2**.

A. Contexto del estudio

Las historias de usuario y los modelos PN se utilizaron durante el desarrollo de once proyectos que se encuentran en la fase de análisis, con la participación de 62 analistas que trabajan en el Departamento de Construcción de Software del Servicio Nacional de Aprendizaje [SENA] – Centro Latinoamericano de Especies Menores [CLEM], regional Valle, y 25 involucrados de diferentes áreas en las que se implementarán los proyectos. Los proyectos son de diversa complejidad y se desarrollan en diversos lenguajes de programación (e.g., Java, PHP, JavaScript), y se desarrollan, básicamente, en las áreas de Talento Humano, Puntos de Atención a Usuarios, Ingreso de Personal y Gestión de Inventarios, entre otros. En el proceso de evaluación se contó con analistas y desarrolladores de sistemas con nivel académico: ochenta estudiantes del Programa de Tecnología en Análisis y Desarrollo de Sistemas de Información del SENA, quienes se encuentran culminando su formación a través del desarrollo de proyectos software; y doce desarrolladores, Ingenieros de Sistemas.

Para dar inicio al diseño de la investigación se capacitó a los analistas participantes en la metodología XP y en la creación de historias de usuario, modelos PN y BPMN. Para la elicitación de requisitos con las historias de usuario se trabajó con una plantilla propuesta Qasaimch (2011). Como ejemplo se toma una de las historias de usuario y un modelo de PN, definidos en el proyecto A. La **TABLA 3** muestra la historia de usuario creada para uno de los requisitos (R1) del Proyecto A, el cual consiste en registrar el ingreso de una persona, así como sus elementos personales si los posee. Luego, en la Figura 3 se ilustra el mismo requisito especificado en un modelo de PN, donde se muestra como el guarda de seguridad: consulta la persona que ingresa y si existe ve su información y pasa a registrar sus elementos personales; consulta el elemento personal y si existe

achieves the best result. Metrics not considered are related with the ease of maintenance and financial status, which are not relevant to the objective. **TABLE 2** describes the metrics applied to the eleven projects, which used either user stories or BPN models as a strategy.

The target of the metrics in this research is to identify which one best helps to improve understanding between the involved parties and analysts, from the starting point of the number of approved requirements and the time each strategy implementation takes in the stages of planning and iteration, as well as exploration. The results reflect that there is an improvement in the final product quality and the productivity of the development team once the requirements are clearly elicited by the involved individuals.

V. Validation and final results

In order to analyze the questions under consideration (with the aim of comparing the quality and quantity of collected data on each of them), the analysts used focus groups of the involved individuals after implementing either the user stories or the BPN models. The focus groups were used as a technique for data collection through a group interview around a specified theme (Escobar & Bonilla-Jimenez, 2009); its main purpose is to bring to light inquiries and reactions from both the involved individuals and the analysts. Following the focus groups study (as a tool for results analysis and evaluation of the effects of the use of the proposed strategy against user stories), case studies were used in order to research an individual entity or tendency for a fixed period. Case studies help industry to assess the benefits of methods and tools, as well as providing an effective way to ensure that changes in a process return the expected results. By definition, case studies are executed in real environments, so they provide a high grade of realism without depending on the level of control.

Both strategies (user stories vs BPN) were applied at the same time on the projects selected for the solution strategy validation, and the metrics described in **TABLE 2** were applied.

A. Study context

User stories and BPN modes were used in the development of eleven projects at the analysis stage with contributions from 62 analysts who work at the Department of Software Construction of the National Learning Service (Servicio Nacional de Aprendizaje) [SENA] – Latin-American Center for Minor Species (Centro Latinoamericano

Table 3. Template for user story case for Project A on R1 / Tabla 3. Plantilla de historia de usuario empleada en el proyecto A para R1

Title: Project A – R1. Register Personal Entry / Título: Proyecto A – R1. Registrar Ingreso de Personal			
Story Review / Revisión histórica			
Rev.	Change Description / Descripción del cambio	Author / Autor	Date / Fecha
Reference Documents / Documentos de referencia			
Document Number: / Documento Número:	Title: / Título:		
1	Security Personnel records– Entry Template / Libro Guarda – Plantilla de Ingreso		
GENERAL INFORMATION / INFORMACIÓN GENERAL			
Actors: / Actores:	Security Personnel / Guarda de Seguridad		
Dependencies: / Dependencias:	Security personnel stored on the system / Guarda Registrado en el sistema		
Priority: / Prioridad:	Low / Baja	<input type="checkbox"/> Medium / Media	<input checked="" type="checkbox"/> High / <input type="checkbox"/>
Story Description / Descripción de la Historia			
<p>When an individual wants to enter the institution, the security officer asks him for his ID. If this is on record, that data is pulled up and the security officer searches for the personal data after a visual check over them. If they are on record, the data is displayed (serial number, name, etc.). Otherwise, the security officer proceeds to register them and to relate them with the individual. Then the security officer stores the time and date of the given entry. / Cuando una persona desea ingresar a la institución, el guarda de seguridad le pide el número de identificación. Si se encuentra registrado, los datos se mostrarán y posteriormente el guarda, previa revisión visual, consultará si los elementos personales a ingresar están registrados. Si lo están se muestra su información (serial, nombre, etc.), si no, el guarda los registrará y los asociará al ingreso de la persona. Luego, el guarda registra el ingreso y el sistema almacena la fecha y hora de dicho ingreso.</p>			
Notes: / Observaciones:			
<p>If the individual is not on record, the security officer will report this issue to the User Contact Center Associate in order to create a record. / Si el personal no está registrado, el guarda comunicará esta novedad al encargado de Atención al Cliente Interno para que éste realice su registro.</p>			

de Especies Menores) [CLEM], Valle office and 25 involved individuals from different areas where the projects would be implemented. The projects were of varying complexity and were developed in several programming languages (e.g., Java, PHP, JavaScript), and covered such areas as Human Resources, User Information Slots, Employee Entrance, Stock Management, among others. During the evaluation process, there were on hand analysts and systems developers on an academic level: 80 students on technical courses in Analysis and Development of Information Systems from SENA, who are finishing their studies through the development of software projects; and twelve systems engineers as developers.

In order to start the design for the research, participating analysts were instructed in XP methodology and the creation of user stories, BPN models and BPMN. For requirements elicitation with user stories, the template proposed by Qasaimieh (2011) was used. For instance, one user story and a BPN model were taken and declared in Project A. TABLE 3 displays the user story created for one of the requirements (R1) for Project A, which consists of creating a register of an individual's entry to a building, as well as personal details if he is carrying any. Then, FIGURE 3 shows the same requirement written in a BPN model, where the security officer at the entrance

ve su información (si no, puede crearlos). Una vez asociados los elementos personales se registra el ingreso con hora y fecha.

B. Método

Los datos se recolectaron a partir de: grupos focales con los analistas y stakeholders, y plantillas para la recolección de métricas a través de encuestas. Las encuestas se realizaron con los analistas que participaron en cada proyecto al finalizar el uso de cada estrategia (TABLA 4). En cada proyecto participaron los stakeholders que conocen los procesos y sus falencias.

C. Resultados de las métricas

Prueba de métrica 1: tiempo de entrega de un requisito tras su petición

Para evaluar la métrica 1 con cuatro de los once proyectos se aplicaron las dos estrategias y al finalizar el proyecto se calculó el promedio (\bar{X}) de entrega por requisito en las iteraciones programadas. Los resultados obtenidos se presentan en la TABLA 5.

La FIGURA 4 muestra el comportamiento de M1 en cada una de las estrategias y permite apreciar que el tiempo promedio de entrega de requisitos elicitados con PN es menor que con historias de usuario (entre un 4% y 20% menos, con una media de 12%). En este caso, la métrica establece que el modelo PN le permite a los analistas especificar con mayor claridad los requisitos de los stakeholders, generando menos validaciones con el cliente para aclarar información que en la historia de usuario no se había especificado. Este tiempo de nuevas entrevistas genera retraso en la implementación del requisito y pérdida de tiempo en la codificación que posiblemente debe ser modificada por información incompleta o ambigua.

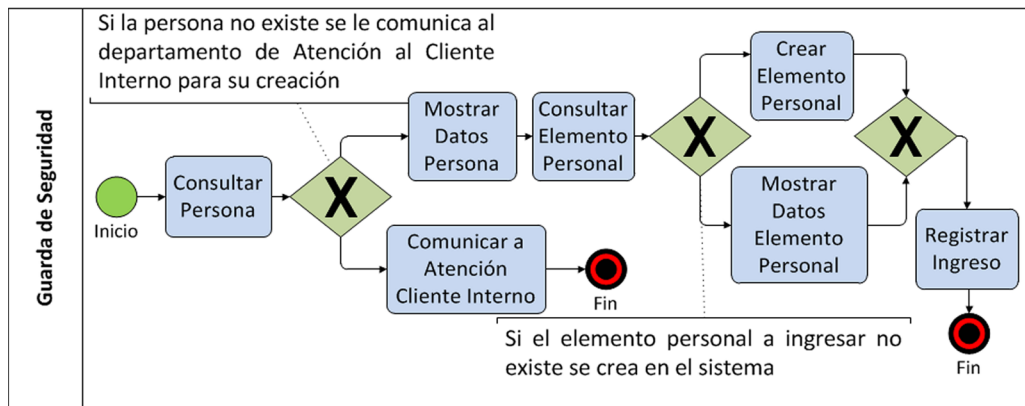


Figure 3. BPN model for R1 at project A / Figura 3. Modelado de PN para R1 del proyecto A

Prueba de métrica 2: requisitos completados en la iteración

Esta métrica se aplicó sobre una muestra de tres proyectos, en los cuales se usó las historias de usuario y los modelos PN para elicitar los requisitos. Los datos hallados se muestran en la TABLA 6.

La FIGURA 5 muestra el comportamiento de M2 en cada una de las estrategias y permite apreciar que el número promedio de requisitos completados/aprobados con PN es mayor que con historias de usuario (entre 20% y 38% más, con un promedio de 32%), lo que permite deducir que al ser más ambigua la especificación de los requisitos con historias de usuario, después de la elicitación aún quedan aspectos o elementos en los requisitos por definir o aclarar con los involucrados. La historia de usuario, por sí misma, no es una estrategia que permita definir de manera completa y clara los requisitos con los involucrados, ocasionando que estos no aprueben su implementación y se deban modificar en las siguientes iteraciones, aumentando el tiempo de entrega, como se evidenció también en la M1.

Prueba de métrica 3: cambios incorporados y requisitos añadidos sobre el alcance inicial del proyecto

La tercera métrica analizada fue aplicada sobre una muestra de cuatro proyectos, en los cuales se contabilizó el número de cambios incorporados a los requisitos o los requisitos añadidos a todo el proyecto. Los datos hallados se muestran en la TABLA 7.

La FIGURA 6 muestra el comportamiento de M3 en cada una de las estrategias y permite apreciar que el número promedio de cambios incorporados a los requisitos que fueron elicitados con PN es menor que con las historias de usuario (entre un 12% y 46% menos, con un promedio de 23%). Los stakeholders quedan con mayor cantidad de dudas cuando se elicitan los requisitos con historias de usuario, lo que genera cambios sustanciales al momento de entregar la implementación o realizar validaciones. Lo contrario sucedió cuando se elicató con modelos PN, los cambios en los requisitos eran

searches for the individual who is trying to enter, and in the case of finding him, proceeds to look for his information and personal data, and after searching for them proceeds to create a record of them if they are not found. Once the personal information is registered and logged, the date and time are also recorded.

tion is registered and logged, the date and time are also recorded.

B. Method

Data was collected from focus groups with analysts and *stakeholders*, and templates for metrics were collected using surveys. The given surveys were performed with analysts who contributed to each project after using each strategy (TABLE 4). *Stakeholders*, who knew about the processes and their failures, supported each project.

C. Metrics results

Metric 1 test: delivery time for a requirement after request

In order to evaluate metric 1, in four of the eleven projects both strategies were applied, and at the end of the project the delivery time average (\bar{x}) was calculated per requirement on the scheduled iterations. TABLE 5 displays the results.

Table 4. Number of projects and analysts by metric / Tabla 4. Número de proyectos y analistas por métrica

Metric / Métrica	Projects / Proyectos	Analysts (#) / Analistas (#)		Involved individuals (#) / Involucrados (#)
		User Stories / Historias de usuario	BPN Models / Modelos PN	
M1		2	2	2
	B	2	2	2
	C	4	3	3
M2	D	4	3	3
	E	6	6	2
	F	6	6	5
M3 and M4	G	6	6	4
	H	4	4	3
	I	4	4	3
	J	4	4	3
		4	4	3

FIGURE 4 displays the behavior for M1 in each of the strategies and shows that the average delivery time for the elicited requirements using BPN is less than for User Stories (around 4% to 20% less, with an average of 12%). The metric in this case indicates that the BPN model helps the analysts to describe the stakeholder requirements even more clearly, by creating fewer validations with the client in order to obtain more information that is not provided by the user story. The time taken for additional meetings leads to delays in the requirement implementation and a waste of time on the code, which needs to be changed because of incomplete or ambiguous data.

Metric 2 test: Completed requirements at iteration

This metric was applied on a sample composed of three projects where both user stories and BPN models were used in order to elicit requirements. TABLE 6 shows the data obtained.

FIGURE 5 displays the behavior for M2 under each of the strategies and shows that the average number of completed/approved requirements with BPN is greater than when using User Stories (around 20% to 38% greater, with an average of 32%), which makes clear that due to the ambiguous requirements specification using User Stories, there are still elements that need to be described or clarified by the involved parties. The user story is not a strategy that allows requirements to be described in a clear and complete way by the involved individuals, which has the result that they disapprove its implementation and need further changes in the next iterations to be performed, taking even more time for delivery, as reflected in M1.

Metric 3 test: Injected changes and added requirements over the initial project approach

The third metric that was analyzed was applied to a sample of four projects, where either the number of changes to the requirements or the requirements added to the whole project were counted. TABLE 7 displays the data obtained.

FIGURE 6 displays the behavior for M3 of each of the strategies and shows that the average number of injected changes to the elicited requirements using BPN is less than when using User Stories (around 12% to 46% less, with an average of 23%). The stakeholders continue to have even more questions with user stories, which results in critical changes at the intended time of delivery

Table 5. Analyzed project data at M1 / Tabla 5. Datos para proyectos analizados en M1

Project / Proyecto	Average delivery time (weeks) for an elicited requirement / Tiempo (semanas) promedio de Entrega de un requisito elicitado	Effectiveness improvement using BPN instead of US (%) / Mejora en efectividad usando PN en lugar de HU
A	2.5	20
B	1.7	12
C	3.0	13
D	2.6	4
\bar{x}	2.5	12

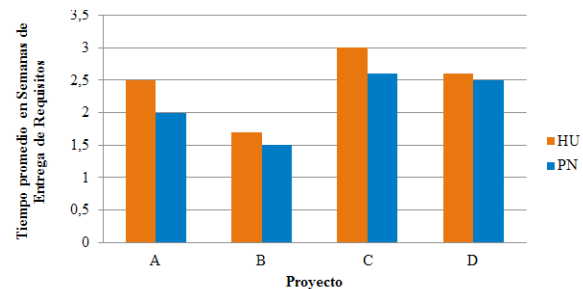


Figure 4. Metric 1. Average delivery time (weeks) for a requirement after its request using BPN models and User Stories / Figura 4. Métrica 1. Tiempo promedio de entrega (semanas) de un requisito tras su petición con modelos de PN e historias de usuario

Table 6. Analyzed projects data at M2 / Tabla 6. Datos para proyectos analizados en M2

Project / Proyecto	Completed requirements average / approved by iteration, elicited / Promedio de requisitos completados / aprobados por iteración elicitados.	Effectiveness improvement using BPN instead of US (%) / Mejora en efectividad usando PN en lugar de HU
E	10	38
F	12	20
G	8	38
\bar{x}	10	32

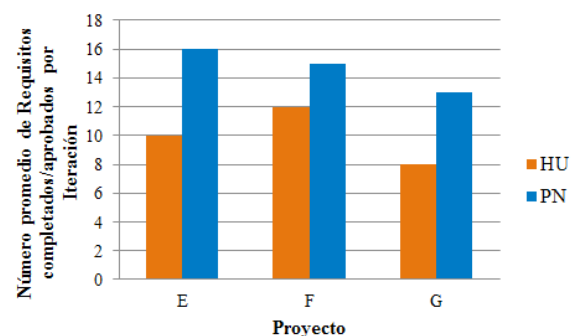


Figure 5. Metric 2. Completed/approved requirements average by iteration using BPN models and User Stories / Figura 5. Métrica 2. Promedio de requisitos completados/aprobados por iteración con modelos de PN e historias de usuario

Table 7. Data for analyzed projects at M3 / *Tabla 7. Datos para proyectos analizados en M3*

Project / Proyecto	Completed requirements average / approved by iteration, elicited / Promedio de cambios incorporados a requisitos		Improvements BPN vs US / Mejoras PN vs HU	Added project requirements / Requisitos añadidos al proyecto		Improvements BPN vs US / Mejoras PN vs HU
	US / con HU	BPN Models / con modelos PN		US / con HU	BPN Models / con modelos PN	
H	2.5	2.2	12%	15	14	7%
I	3.6	3.0	17%	18	18	0%
J	4.5	3.8	16%	21	20	5%
K	3.7	2.0	46%	12	11	8%
\bar{X}	3.58	2.75	23%	16.50	15.75	5%

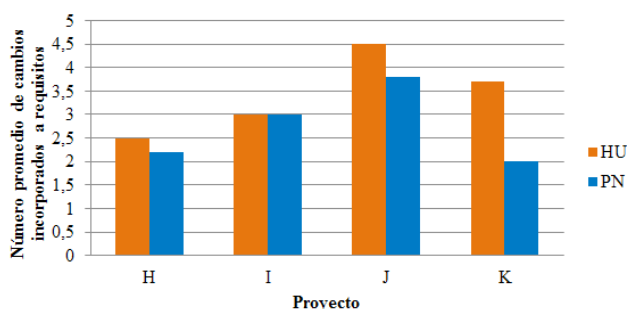


Figure 6. Metric 3. Average number of injected requirements with BPN and User Stories / *Figura 6. Métrica 3. Promedio de cambios incorporados a requisitos con modelos de PN e historias de usuario*

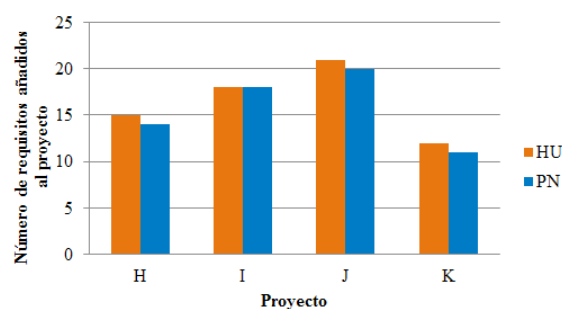


Figure 7. Metric 3. Number of requirements added to the project using BPN models and User Stories / *Figura 7. Métrica 3. Número de requisitos añadidos al proyecto con modelos de PN e historias de usuario*

menores y muchos de los detalles y restricciones asociadas a las funcionalidades especificadas ya se habían resuelto en la especificación de las necesidades, en la fase de exploración.

La FIGURA 7 además muestra el número de requisitos que fueron añadidos en el total de iteraciones por proyecto, donde la diferencia entre historias de usuario y modelos PN no es muy notoria, dado que los requisitos nuevos se dan principalmente por los cambios en los procesos y estos son independientes de la estrategia que se use para su elicitación (entre un 0% y un 8% menos, con un promedio de 5% menos en los cuatro proyectos comparados).

Prueba de métrica 4: incidencias (defectos encontrados por el cliente o usuarios del producto) según estado y criticidad

La cuarta y última métrica analizada fue aplicada sobre una muestra de cuatro proyectos, en los cuales se contabilizó el número de defectos encontrados por el stakeholder en cada iteración, con historias de usuario o modelos PN. Esta métrica difiere de la anterior en que los defectos son considerados como errores en la programación del requisito, lo que hace que la implementación no cumpla con la necesidad del cliente, a pesar de que está bien elicitado. Esta métrica permitió medir los cambios que podían ser causados por modificaciones del proceso o por funcionalidades que no habían sido tenidas en cuenta por el stakeholder. Los datos hallados se muestran en la TABLA 8.

La FIGURA 8 muestra el comportamiento de M4 en cada una de las estrategias y permite apreciar que el número promedio de defectos identificados por el stakeholder en los requisitos, es menor

or execution of validations. On the other hand, when the elicitation is done using BPN models, there are only minor changes and many restrictions and details related to specific functionalities have already been resolved at the needs description in the exploration stage.

Moreover, FIGURE 7 shows the number of requirements added on the iterations summary by project, where there is barely any difference between the User Stories and BPN models, since new requirements are given mainly by changes in processes and are independent from the strategy executed for elicitation (around 0% to 8% less, with an average of 5% less in the four compared projects).

Metric test 4: Issues (bugs found by client or product user) based on status and critical level

The fourth and final metric analyzed was applied to a sample of four projects, where the number of issues found by the stakeholder at each iteration, using either User Stories or BPN models, was counted. This metric differs from the previous one as issues are considered as errors in the requirement programming, which means that the implementation does not fulfill the client needs, even though it is well elicited. This metric provides a measurement of the changes that may be caused by either process modifications or functiona-

lities that the stakeholder did not have under consideration. **TABLE 8** displays the data obtained.

FIGURE 8 shows the behavior of M4 for each of the strategies and shows that the average number of issues identified by the stakeholder at the requirements stage is less when they were elicited using BPN models than when using User Stories (around 24% to 50% fewer issues, with an average of 34%). The difference in this metric is more evident in each of the analyzed projects and it is clear that user stories, through their natural language, cause the analyst to misunderstand the given needs, and thus to implement functionalities that do not in the end fulfill expectations.

VI. Discussion

Metrics analysis allows the use of BPN models as a support tool in the process of requirements elicitation that helps to enhance the quantity of completed/approved requirements by 32% compared with the use of user stories. Furthermore, the average number of injected changes to requirements using BPN modes is about 2.75, with an enhancement of 23% against user stories. Another feature evaluated with metrics, when BPN models are included, reflects an improvement in the number of issues with the requirements found by the client after delivery by about 34% compared with user stories. Reducing these issues improves the delivery time, as the requirement is not returned for review, which tightens the story for a new iteration.

Requirements elicitation in agile development software depends heavily on user stories, so there will be issues within the project if they are wrongly defined. Writing user stories in natural language might suggest that there are no mistakes (by mentioning the requirements in a simple way); however, this simplicity may cause problems of understanding and ambiguity, which will be reflected in bugs found by the *stakeholder*, as shown by the use of metrics. Miswriting user stories may cause an incorrect understanding of requirements, wrong formulation of test cases, wrong implementation and consequently delivery refusal at iterations.

VII. Conclusion and future work

Experimentation executed through the proposed solution allows us to conclude that the BPN model as a requirements elicitation tool enhances communication between the analyst and involved individuals, which leads to a rise in the number of approved requirements when

Table 8. Data for projects analyzed at M4 / Tabla 8. Datos para proyectos analizados en M4

Project / Proyecto	Average issues in elicited requirements / Promedio de defectos en requisitos elicitados		Quality improvement using BPN instead of US (%) / Mejora en calidad usando PN en lugar de HU
	US / con HU	BPN Models / con modelos PN	
H	3.4	2.1	38
I	2.6	1.8	31
J	5.0	3.8	24
K	2.6	1.3	50
\bar{X}	3.40	2.25	34

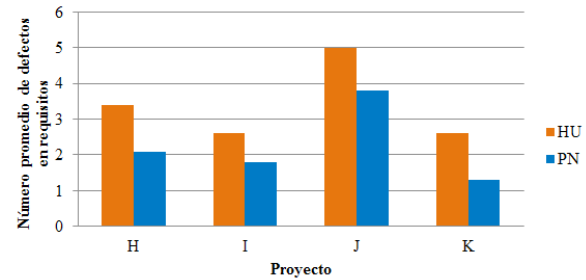


Figure 8. Metric 4. Average number of issues on requirements elicited with BPN models and User Stories / Figura 8. Métrica 4. Número promedio de defectos en requisitos elicitados con modelos de PN e historias de usuario

cuando fueron elicitados con PN, que con historias de usuario (entre 24% y 50% menos defectos, con un promedio de 34%). En esta métrica la diferencia es más notoria entre cada uno de los proyectos analizados y se evidencia que las historias de usuario, a través del lenguaje natural, ocasionan una mala interpretación de la necesidad por parte del analista, quien implementa funcionalidades que al final no cumplen con lo solicitado.

VI. Discusión

El análisis de las métricas permite establecer que el uso de modelos PN como apoyo en el proceso de elicitación de requisitos permite aumentar la cantidad de requisitos completados / aprobados en un 32%, respecto del uso de historias de usuario. Además, el número promedio de cambios incorporados a los requisitos con modelos PN es de 2.75 con una mejora del 23% respecto de las historias de usuario. Otro aspecto importante evaluado con las métricas en la inclusión de los modelos PN evidenció una mejora en el número de defectos encontrados por el cliente en los requisitos después de la entrega, que fue del 34% respecto del uso de las historias de usuario. Estos defectos disminuyen el tiempo de entrega sin tener que devolver el requisito, ajustando la historia, para una nueva iteración.

La elicitación de requisitos en el desarrollo ágil de software depende en gran medida de las historias de usuario, por lo que si se cometen errores al definir las, se ocasionan problemas a lo largo del desarrollo. Redactar en lenguaje natural las historias de usuario podría indicar que no se cometen errores –al expresar requisitos de una forma tan simple–, sin embargo,

esta simplicidad puede ocasionar problemas de entendimiento y ambigüedades que son reflejadas en los defectos que puede encontrar el *stakeholder*, como se evidencio con el uso de las métricas. Los errores cometidos al escribir las historias de usuario pueden ocasionar: mal entendimiento de los requisitos, mala formulación de casos de pruebas, mala implementación y, en consecuencia, el rechazo de los entregables en una iteración.

VII. Conclusiones y trabajo futuro

La experimentación realizada a través de la estrategia propuesta permite concluir que el uso de modelos PN como herramienta para la elicitación de requisitos permite mejorar la comunicación entre el analista y los involucrados, lo que se evidencia con el aumento en el número de requisitos que fueron aprobados al usar esta estrategia. Este aumento en el número de requisitos aprobados, a su vez reduce el tiempo de validación en la fase de planificación y genera menor tiempo en el desarrollo y entrega de las funcionalidades programadas en el *Release Plan*. La reducción del tiempo en fases posteriores a la elicitación se convierte en una oportunidad que mejora la productividad del equipo de desarrollo y reduce el tiempo de cada iteración logrando así satisfacer las necesidades de los involucrados, con requisitos de calidad. También se logra evidenciar una mejor interpretación y comprensión de los requisitos que han sido elicitados a través de modelos de PN. A pesar de que las historias de usuario permiten la aprobación de requisitos, este número es menor a los requisitos que son aprobados cuando se especifican con modelos de PN.

Como trabajo futuro se incluye: validar si la cantidad de programas de prueba (*spikes*) que se generan al usar PN es menor a la que se genera con historias de usuario en la fase de planificación; validar el tiempo de desarrollo de los *spikes* con cada estrategia; y explorar el uso de modelos PN en otras metodologías ágiles, como *Scrum*. **ST**

using this strategy. This increase in the number of approved requirements also reduces the validation time in the planning stage and shortens the development delivery time for the functionalities scheduled in the *Release Plan*. The decrease of time required for the stages after elicitation becomes an opportunity to improve the development team's overall productivity and reduces the time for each iteration, which helps to improve the satisfaction of the needs of the involved individuals as well as the quality requirements. Also, there is an improvement in the comprehension and understanding of the elicited requirements using BPN models. Although user stories help with the approval of requirements, this number is less than those approved when using BPN models.

In future work it is intended to validate whether or not the quantity of test programs (*spikes*) generated when using BPN is less than that created with user stories in the planning stage; to validate the development time for *spikes* under each strategy; and to explore the use of BPN models across other agile methodologies such as *Scrum*. **ST**

References / Referencias

- Al-Zoabi, Z. (2008). Introducing discipline to XP: Applying PRINCE2 on XP projects. *3rd International Conference on Information and Communication Technologies: From Theory to Applications, ICTTA*.
- Beck, K. (2000). *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional.
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... Thomas, D. (2001). *Manifiesto por el desarrollo Ágil de Software*. Retrieved from <http://www.agilemanifesto.org/>
- Blom, M. (2010). Is Scrum and XP suitable for CSE Development? *Procedia Computer Science*, 1, 1511–1517.
- Chinosi, M., & Trombetta, A. (2012). BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 34(1), 124–134.
- Christensen, H. B. (2014). Analysis and design of software ecosystem architectures – Towards the 4S telemedicine ecosystem. *Information and Software Technology*, 56, 1476–1492.
- Domann, J. (2014). An agile method for multiagent software engineering. *Procedia Computer Science*, 32, 928–934.
- Dorigan, J. A., & Miranda de Barros, R. (2014). A process model for standardization and increase in the requirements quality. *Latin America Transactions, IEEE (Revista IEEE America Latina)*, 12(8), 1502-1507.
- Escobar, J., & Bonilla-Jimenez, F. I. (2009). Grupos focales: una guía conceptual y metodológica. *Cuadernos Hispanoamericanos de Psicología*, 9(1), 51–67.
- Fair Isaac Corporation (2010). *Enhancing BPM with business rules* [white paper]. Retrieved from: http://www.softwareag.com/il/Images/FICO_SAG_BPM_BRMS_2646WP_tcm98-76299.pdf
- Jaqueira, A., Lucena, M., Aranha, E., Alencar, F., & Castro, J. (2013). Using i* models to enrich user stories objectives of the research. *Proceedings of the 6th International i* Workshop (iStar 2013)*, CEUR, (Vol. 978, pp. 55-60).
- Joskowicz, I. J. (2008). *Reglas y prácticas en eXtreme programming* [technical report]. Vigo, Spain: Universidad de Vigo.

- Kazi, L., Radosav, D., Nikolic, M., & Chotaliya, N. (2011). Balanced scorecard framework in software project monitoring. *Journal of Engineering Management and Competitiveness (JEMC)*, 1(1-2), 51-56.
- Losada, B. M., Urretavizcaya, I. F.-C. (2013). A guide to agile development of interactive software with a "User Objectives"-driven methodology. *Science of Computer Programming*, 78, 2268-2281.
- Mukker, A. R., Singh, L., & Mishra, A. K. (2014). Systematic review of metrics in software agile projects. *Compusoft*, 3(2), 533-539.
- Newkirk, J. (2002). Introduction to agile processes and extreme programming. *Proceedings of the 24th International Conference on Software Engineering - ICSE '02*, 695. doi:10.1145/581441.581450
- Object Management Group [OMG]. (2006). *Business process modeling notation specification*. Retrieved from: http://www.omg.org/bpmn/Documents/OMG_Final_Adopted_BPMN_1-0_Spec_06-02-01.pdf
- Ordoñez, H., Escobar, A., Velandia, D., Cobos, C., Ordoñez, A., & Segovia, R. (2015). An impact study of business process models for requirements elicitation in XP. In O. Gervasi, B. Murgante, S. Misra, M. L. Gavrilova, A. M. Alves Coutinho Rocha, C. Torre, ... B. O. Apduhan (Eds.), *Computational Science and its Applications - ICCSA 2015* (pp. 298-312). Springer International Publishing. doi:10.1007/978-3-319-21404-7_22
- Ottensooser, A., Fekete, A., Reijers, H. A., Mendling, J., & Menictas, C. (2012). Making sense of business process descriptions: An experimental comparison of graphical and textual notations. *Journal of Systems and Software*, 85(3), 596-606. doi:10.1016/j.jss.2011.09.023
- Paternoster, N., & Giardino, C. (2014). Software development in startup companies: A systematic mapping study. *Information and Software Technology*, 56(10), 1200-1218.
- Pino, F. J. (2010). Assessment methodology for software process improvement in small organizations. *Information and Software Technology*, 52, 1044-1061.
- Pohl, K. (2010). *Requirements engineering: fundamentals, principles and techniques*. Berlin-Heidelberg, Germany: Springer Verlag.
- Qasaimeh, M. (2011). Extending extreme programming user stories to meet ISO 9001 formality requirements. *Journal of Software Engineering and Applications*, 4 (11), 626-638.
- Schneider, F., Naughton, H., & Berenbach, B. (2012). A modeling language to support early lifecycle requirements modeling for systems engineering. *Procedia Computer Science*, 8, 201-206. doi:10.1016/j.procs.2012.01.043
- Stephens, B. M. (2002). *The case against extreme programming* [on-line]. Retrieved from <http://www.lyee-project.soft.iwa-te-pu.ac.jp/en/workplan/kb/library/xp/documents/papers/s2001.pdf>
- Suaza, K.V. (2013). *Definición de equivalencias entre historias de usuario y especificaciones en UN - LENCEP para el desarrollo ágil de software* [Thesis]. Universidad Nacional de Colombia: Medellín, Colombia.
- Thiyagarajan, P. S. & Verma, S. (2009). A closer look at extreme programming (XP) with an onsite-offshore model to develop software projects using XP methodology. In *Lecture Notes in Business Information Processing* (Vol. 16, pp. 166-180). Berlin-Heidelberg: Springer-Verlag. doi: 10.1007/978-3-642-01856-5_12
- Tong, K. I. (2010). Managing software projects with user stories. In *Essential Skills for Agile Development* (pp. 217-252). Xangai, Macau: Macau Productivity and Technology Center (CPTTM).
- Wong, P. Y., & Gibbons, J. (2011). Formalizations and applications of BPMN. *Science of Computer Programming*, 76(8), 633-650.
- Zheng, G. (2012). *Implementing a business process management system applying Agile development methodology: A real-world case study* [Thesis]. Erasmus Universiteit Rotterdam: The Netherlands

CURRICULUM VITAE

Andrés Felipe Escobar Villada Systems Engineer (Universidad del Valle, Cali-Colombia); Software Development Process Specialist (Universidad San Buenaventura de Cali); and current student on Master in Software Engineering (Universidad San Buenaventura de Cali). At present, he works as a Development Engineer at Arquitectsoft S.A.S. His research areas of interest include: software engineering and mobile & web application development. / Ingeniero de Sistemas (Universidad del Valle, Cali-Colombia); Especialista en Procesos para el Desarrollo de Software (Universidad San Buenaventura de Cali); y estudiante de Maestría en Ingeniería de Software (Universidad San Buenaventura de Cali). Actualmente se desempeña como Ingeniero de Desarrollo en Arquitectsoft S.A.S. Sus áreas de interés en investigación incluyen: Ingeniería de Software y el desarrollo de aplicaciones en la Web y para dispositivos móviles.

Diana Lorena Velandia Vanegas Systems Engineer (Universidad del Valle, Cali-Colombia); Software Development Process Specialist (Universidad San Buenaventura de Cali); and current student on Master in Software Engineering (Universidad San Buenaventura de Cali). At present, she works as an instructor at the Learning National Service [SENA]. Her research areas of interest include: software engineering, software development methodologies and web applications. / Ingeniera de Sistemas (Universidad del Valle, Cali-Colombia); Especialista en Procesos para el Desarrollo de Software (Universidad San Buenaventura de Cali); y estudiante de Maestría en Ingeniería de Software (Universidad San Buenaventura de Cali). Actualmente se desempeña como instructora de informática en el Servicio Nacional de Aprendizaje [SENA]. Sus áreas de interés en investigación incluyen: Ingeniería de Software, metodologías de desarrollo de software y aplicaciones en la Web.

Hugo Ordoñez Ph.D. Telematics Engineering (Universidad del Cauca, Popayán-Colombia). He is a professor at the Engineering School at Universidad de San Buenaventura de Cali and a member of the research group on Telematics Engineering at Universidad del Cauca. His research areas of interest are: business process mining, data recovery, web documents clustering and software engineering. / Ph.D., en Ingeniería Telemática (Universidad del Cauca, Popayán-Colombia). Es profesor titular de la Facultad de Ingeniería de la Universidad de San Buenaventura de Cali, e investigador del grupo de Ingeniería Telemática de la Universidad del Cauca. Sus áreas de interés en investigación son: la minería de procesos de negocio, la recuperación de información, el clustering de documentos Web y la Ingeniería de Software.

Carlos Cobos Ph.D. Computer Science and Systems Engineering. He is a professor at the Systems department in the school of Electronics and Telecommunications and leader at the R+D group of Information Technology (GTI) at Universidad del Cauca (Popayán-Colombia). His research areas of interest are: information recovery, data mining, systems optimization through meta-heuristics and e-learning. / Ph.D., en Ingeniería de Sistemas y Computación. Es profesor titular del Departamento de Sistemas en la Facultad de Ingeniería Electrónica y Telecomunicaciones y Coordinador del Grupo de I+D en Tecnologías de la Información (GTI) de la Universidad del Cauca (Popayán-Colombia). Sus áreas de interés en investigación son: la recuperación de información, la minería de datos, la optimización de sistemas complejos usando meta-heurísticas y la educación en línea.